Motion Tracking of Infants in Risk of Cerebral Palsy

Olsen, Mikkel Damgaard

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

Motion Tracking of Infants in Risk of Cerebral Palsy

Mikkel Damgaard Olsen

Kongens Lyngby 2015
PHD-2015-393

Technical University of Denmark
Department of Applied Mathematics and Computer Science
Building 324, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253031
Fax +45 45881399
compute@compute.dtu.dk
www.compute.dtu.dk

PHD: ISSN 0909-3192

Summary

Every year 2-3 out of 1000 infants are born with cerebral palsy. Among others, the disorder often affects motor, cognitive and perceptual skills. The disorder is usually detected when the infant is old enough to crawl and walk, i.e. when the infant is 1-2 years old. However, studies show that the infant's movements are affected already in the first year of life and methods exist for assessing these. The methods often involves visual observation and qualitative evaluation of the movements. A more objective measure is desired in order to be able to diagnose cerebral palsy much earlier.

The goal with this thesis is to describe the development of a markerless motion tracking system for infants. Based on data recorded with a low-cost depth sensor, image analysis and mathematical modeling is used to model the infant's body and its movements. Two methods are considered, where the first method is able to do single frame pose estimation, based on simple assumptions on the infant's body. The second method uses an articulated model that incorporates anatomical constraints. Combining the two methods results in a robust motion tracking system for infants.

The results from the motion tracking are used to extract physical features such as velocity and acceleration of the individual body parts. A novel method for estimating scene flow in human motion data is presented, utilizing the results from the motion tracking. A number of examples are given for potential applications for automatic assessment of infant movement. This includes a preliminary study on automatic classification of movements related to cerebral palsy.

The contributions included in this thesis can be divided into two groups. The

first two contributions consider the analysis in order to estimate and track the body of the infants. The remaining contributions consider different motion features derived from the motion tracking results. Both pose and motion features are extracted and used for assessing the infants' motor development.

The presented work is a step closer to automatic motion assessment of infants with focus on early diagnosis of infants with cerebral palsy. Further collaboration with clinicians can result in breakthroughs in the way infants are monitored and assessed during the early years of life.

The main motivation is to be able to assess infants in risk of cerebral palsy based on the previously established connection between infant movement and brain injuries. However, as the data used in this study is recorded simultaneously with the study, the true outcome is not known. Even though some of the included infants were born preterm, none of them have currently been diagnosed with cerebral palsy.

Resumé

Hvert år fødes der 2-3 ud af 1000 børn med cerebral parese. Symptomerne er blandt andet motoriske, kognitive og perceptuelle vanskeligheder. Sygdommen opdages oftest når barnet er i stand til at gå eller kravle, dvs. når barnet er 1-2 år gammel. Forskning viser dog at barnets bevægelser allerede er påvirket i det første år efter fødslen og der eksisterer metoder der kan bedømme barnets bevægelser. Metoden kræver oftest observation og kvalitativ måling af bevægelserne. Derudover spiller observatørens erfaring en væsentlig rolle. Derfor ønskes der en mere objektiv metode, for at kunne diagnosticerer cerebral parese tidligt.

Målet med denne afhandling er at beskrive udviklingen af et system der er i stand til modellere spædbørns bevægelser, uden brug af udstyr der fastgøres på barnet. Vi benytter teknikker inden for billedanalyse og matematisk modellering, på data optaget med et lettilgængeligt dybdekamera, for at kunne detektere og modellere børnenes bevægelser. To forskellige metoder benyttes for at løse dette problem. Den ene metode er i stand til at finde og identificerer hænder, fødder og hoved, ved brug af generelle antagelser omkring menneskets figur. Den anden metode benytter en artikuleret model, som tager højde for kroppens anatomiske begrænsninger. Ved at kombinere de to metoder, er vi i stand til at finde og modellere barnets krop og bevægelser.

Resultaterne benyttes til at udtrække informationer såsom hastigheder og accelerationer af barnets enkelte kropsdele og led. Derudover præsenteres en ny metode til at beregne scene flow i optagelser af mennesker, ved brug af resultaterne fra de ovennævnte metoder. Der gives en række eksempler på hvordan målingerne kan benyttes til automatisk bedømmelse af børns bevægelser. Dette inkluderer blandt andet at detektere bevægelser der er særligt relevante når man

undersøger om spædbørn lider af cerebral parese.

Artiklerne inkluderet i denne afhandling kan opdeles i to kategorier, baseret på deres indhold. De første to artikler beskriver udviklingen af metoderne til at lave bevægelsesanalyse på spædbørn. De resterende artikler omhandler forskellige mål der kan udledes fra bevægelsesanalysen. Dette omfatter blandt andet en analyse af hvordan børns motoriske evner kan bedømmes automatisk.

Resultaterne der præsenteres i denne afhandling er et skridt tættere på et system der er i stand til at automatisk udtrække informationer om børns motoriske udvikling med fokus på tidlig diagnose af børn med cerebral parese. Tæt samarbejde med læger og terapeuter kan føre til gennembrydende resultater inden for tidlig diagnose af motoriske sygdomme.

Projektet er blevet udarbejdet med målet om at kunne diagnosticere cerebral parese tidligere end normalt. Dette baseres på tidligere opdagelser om forbindelsen mellem spædbørns bevægelser og hjerne skader. De data der benyttes er optaget samtidig med studiet og derfor kendes børnenes endelige udfald ikke. Nogle af børnene er født for tidligt, men ingen af dem er til dags dato blevet diagnosticeret med cerebral parese.

Preface

This thesis was prepared at The Image Analysis and Computer Graphics Group at The Department of Applied Mathematics and Computer Science, The Technical University of Denmark. The thesis was submitted to The Technical University of Denmark in partial fulfillment of the requirements for acquiring the Ph.D. degree in engineering. The thesis was carried out with funding from the Ludvig & Sara Elsass Foundation. The project was supervised by associate professor Rasmus R. Paulsen from The Technical University of Denmark and Professor Jens Bo Nielsen from The Helene Elsass Center and The Panum Institute.

The thesis deals with modeling the infant body and movements based on 3D data, with the goal of extracting motion features related to cerebral palsy and motor development. The thesis consists of a summary report and a collection of research papers written during the period 2012–2015, and elsewhere published.

Lyngby, October 2015

Mikkel Damgaard Olsen

Contributions

Papers included in the thesis

[A] **Mikkel Damgaard Olsen**, Anna Herskind, Jens Bo Nielsen, and Rasmus R. Paulsen. Body Part Tracking of Infants. In *22nd International Conference on Pattern Recognition*. Institute of Electrical & Electronics Engineers (IEEE), aug 2014.

[B] **Mikkel Damgaard Olsen**, Anna Herskind, Jens Bo Nielsen, and Rasmus R. Paulsen. Model-Based Motion Tracking of Infants. In *Computer Vision - ECCV 2014 Workshops*, pages 673–685. Springer International Publishing, 2015.

[C] **Mikkel Damgaard Olsen**, Anna Herskind, Jens Bo Nielsen, and Rasmus R. Paulsen. Using Motion Tracking to Detect Spontaneous Movements in Infants. In *Image Analysis*, pages 410–417. Springer International Publishing, 2015.

[D] Gudmundur Einarsson, **Mikkel Damgaard Olsen**, Line H. Clemmensen, and Rasmus R. Paulsen. Scene Flow on Human Motion Data.

[E] **Mikkel Damgaard Olsen**, Gudmundur Einarsson, Jens Bo Nielsen and Rasmus R. Paulsen. Modeling Poses of Infants Using Machine Learning and Motion Tracking

Contributions not included

- Morten Nobel-Jørgensen, Jannik Boll Nielsen, Anders Boesen Lindbo Larsen, **Mikkel Damgaard Olsen**, Jeppe Revall Frisvad, and J. Andreas Bærentzen. Pond of Illusion: Interacting Through Mixed Reality, SIGGRAPH Asia 2013
- **Mikkel Damgaard Olsen**, Anna Herskind, Jens Bo Nielsen, and Rasmus R. Paulsen Assisting Doctors on Assessing Movements in Infants Using Motion Tracking. Developmental Medicine & Child Neurology 2015
- Dal Corso, A., **Olsen, M.**, Steenstrup, K. H., Wilm, J., Jensen, S., Paulsen, R., Eiriksson, E., Nielsen, J., Frisvad, J. R., Einarsson, G., and Kjer, H. M. VirtualTable: A Projection Augmented Reality Game

Acknowledgements

I would like to thank my two supervisors, Associate Professor Rasmus R. Paulsen and Professor Jens Bo Nielsen for supporting me during the project.

I would also like to thank the Ludvig & Sara Elsass Foundation for funding the project, as well as The Helene Elsass Center that I have considered as my second workplace. I have really enjoyed being at the center, both working on my own project, but also participating in other projects, such as the Caretoy project. Even though I owe my thanks to everyone at the center, I would especially like to thank Anna Herskind and Anina Ritterband Rosenbaum, whom been the main persons recruiting most of the infants included in the analysis. I would also like to thank Anna Herskind, Camilla Børregard Voigt, Line Zacho Petersen and Maria Willerslev Olsen for helping with the general movement assessment. I would also like to thank Peder Esben Bilde for our motivating, enthusiastic discussions and introducing me to the wonderful world of gadgets.

I would like to thank all my colleagues in the Image Analysis and Computer Graphics Group at DTU Compute for creating a professional and social environment. I would especially like to thank Martin Kjer, Jannik Boll Nielsen, Morten Nobel-Jørgensen and Gudmundur Einarsson for our discussions during the project, regarding topics on anatomy, the Kinect, data analysis and Cuda programming a.o.

For helping preparing the thesis, I would like to thank Rasmus R. Paulsen, Gudmundur Einarsson, Martin Kjer, Andreas Olsen and Nicoline J. Handschuh.

I also owe APA (Afspændingspædagogisk Aftenskole) my thanks, as they helped

me get in contact with some of the families that I visited in the first part of the project.

Finally, I would like to thank all the families that allowed me to record their adorable infants. This both includes the families that participated in the Caretoy project and the GM/EMG/EEG project at the Helene Elsass Center, as well as the families that allowed me to visit them at home and record their infants. Obviously, without the help from these families this project would not have been possible.

Contents

Part I

Summation

Chapter 1

Introduction

1.1 Motivation

In Denmark, approximately 150 infants are born every year with the motion disorder Cerebral Palsy (CP), corresponding to 2-3 out of 1000 infants [84]. Similar numbers can be observed in other countries [30]. The disorder is caused by damage to the immature brain and in most cases, this injury happens before or in relation to birth. The injury can also occur after birth e.g. due to physical damage to the head. The disorder affects the person's movements and the person will live with the disorder for the rest of his/her life. In addition to the motion-related effects, the disorder can also affect social, cognitive and perceptual skills. Due to the symptoms of the disorder, CP is rarely detected in the early years of life. The disorder is usually suspected due to abnormal movements, when the infant/child is old enough to crawl and walk.

One aspect of CP, which has lately received increased focus, is early intervention. It is known that the brain develops and changes through life and that the development is crucial in the early years after birth. It is thus important that the diagnosis is given as early as possible in order to start intervention. Several methods for early diagnosis of CP exist and especially the general movement assessment method shows strong predictive results [18]. The methods are based on assessing infants' movements in order to detect abnormal movement patterns.

These assessment methods often require that the doctor or clinician observes and scores the infant, based on the infant's ability to meet certain motor milestones or based on the variability/complexity in the infant's movements. Milestones can e.g. be describes as the infant's ability to coordinate multiple body parts or roll from side to side. These milestones are easily observed and can simply be described from one person to another. But when is an infant's movements complex and variable? This measure cannot be as easily explained and one person's understanding of complexity might be based on many years' of practical experience. Due to this subjective understanding of infant's movements, more quantitative measures are desired to extract information of the infant's motor skills.

One solution to this problem is motion tracking or motion capture. This technique has been used for years in e.g. movie and video game production, as well as motion analysis of sports athletes. However, the techniques often require time-consuming preparations or a controlled environment; two things that cannot be taken for granted when working with infants. The techniques can thus not always be applied directly to infants. The goal with this thesis is thus to adapt motion tracking techniques to infants without the need of complex setups that require time-consuming calibration procedures. The goal is not to replace the clinicians, but instead to develop a tool to assist the clinician during the assessment and even detect and point out relevant movement behaviors. Early diagnosis of CP is one application and this has been the main motivation of the project.

1.2 Thesis Overview

The thesis is composed of two parts, where part I describes the work carried out during the PhD study and part II is constructed from the papers published or submitted during the PhD study. It should be noted that there are overlapping descriptions between part I and part II.

1.2.1 Part I

The first part summarizes the work of the PhD study and is intended to be read in chronological order. However, based on the reader's existing knowledge in the field of CP and motion tracking, the chapters 2 and 3 can be skipped.

- **Chapter 1** introduces and motivates the thesis and gives an overview

of the different notations used throughout the thesis. In addition, the included papers are summarized briefly.

- **Chapter 2** describes the clinical background for the study, introducing the motion disorder CP and how people with CP are affected during their life. The chapter also considers early intervention of infants and how CP can be detected early in life, based on clinical methods.
- **Chapter 3** lists different systems for measuring motion and describes how motion can be modeled.
- **Chapter 4** summarizes different techniques used for markerless motion tracking and covers the research done within the field of motion tracking in general and motion tracking of infants, with focus on early prediction of CP.
- **Chapter 5** describes the setup for recording the infants' movements and how the data are processed, in order to remove noisy/undesired measurements.
- **Chapter 6** explains the different methods for doing motion tracking of infants. This both includes a method for roughly estimating the locations of body extremities, as well as a more anatomically correct approach.
- **Chapter 7** focuses on feature extract and further post processing of the results from the motion tracking. The concept of scene flow is explained, which is a dense description of motion Three-dimensional (3D) data.
- **Chapter 8** considers different approaches for classifying the poses and movements of infants, based on the features explained in chapter 7. Different methods for assessing the infants' abilities to meet certain motor milestones are considered. Furthermore, a preliminary study on automatic detection of movements related to CP is conducted.
- **Chapter 8** concludes the thesis and considers future goals and suggestions for improvements.

1.2.2 Part II

Here, the papers included in the thesis are summarized. This is done to give the reader an overview of the goals and outcomes of the papers.

- **Paper A: Body Part Tracking of Infants**

  - **Goal**: The goal with this paper is to estimate the 3D locations of an infant's extremities, based on data obtained with a depth sensor.
  - **Outcome**: Using depth and color images as input for image analysis and graph theory, the paper describes an approach for estimating the locations of an infant's extremities. The approach determines the locations of the head, hands and feet by identifying the five points furthest away from the center of the infant's stomach. The metric is based on the shortest geodesic distance along the surface of the body object. These geodesics are estimated using the Dijkstra's shortest path algorithm applied to the depth data.

- **Paper B: Model-Based Motion Tracking of Infants**
  - **Goal**: Develop a system for doing motion tracking of infants, incorporating anatomical constraints.
  - **Outcome**: Using 3D structures for representing different body parts of a human person, optimization is used to fit an articulated 3D model to depth data. The model is connected by a skeleton structure and parameters explain position, size and orientation of the different body parts. The method is able to estimate and track the movements of infants.

- **Paper C: Using Motion Tracking to Detect Spontaneous Movements**
  - **Goal**: The goal with this paper is to generate features that describe spontaneous movements in infants, such as kicking and punching.
  - **Outcome**: Based on an infant motion tracking system we extract features describing the movements of infants' body parts. Based on different classifiers, we classify sequences of motion data containing spontaneous movements.

- **Paper D: Scene Flow on Human Motion Data**
  - **Goal**: To give a review of the existing methods on scene flow estimation and compare their performance on data containing human motion.
  - **Outcome**: The paper summarizes the concepts of scene flow, which is the 3D analog of optical flow. A description of the existing methods are given and tested on our own human motion dataset. Using synthesized data, we are able to compare the estimated scene flow with the ground truth and estimate the error. We furthermore describe a novel method for scene flow estimation of human motion data, based on the results from a motion tracking system.

- **Paper E: Modeling Poses of Infants Using Machine Learning and Motion Tracking**
  - **Goal**: To quantitatively model and analyze the variation in infants' poses and relate this to methods for assessing infant motor development.
  - **Outcome**: Using an infant motion tracking system, a database of infants and their poses is obtained. From this, the paper uses different methods from machine learning to extract the most common poses of infants and model the variation. The analysis is done on different age-groups and the results relate to the different milestones considered in clinical assessment techniques.

1.2.3 Notation

- The notation $||\bullet||$ is reserved for the 2-norm $||\bullet||_2$. Other norms are indicated with the appropriate subscript.
- When referencing to the Kinect sensor, this is by default version 1. In the case of Kinect version 2, this will be written explicitly.
- Vectors are written as bold lower case letters and are always considered as column-vectors
$$\mathbf{a} \in \mathbb{R}^{k\times 1}$$
Referencing to an element in a vector is done using a single subscript number, i, referencing to the ith element in the vector, e.g. $\mathbf{a}_i$. The number of elements in a vector is written as $k = |\mathbf{a}|$.
- Matrices are written as bold upper case letters
$$\mathbf{A} \in \mathbb{R}^{m\times n}$$
If matrices are used to store multivariate observations, each row is a variable and each column is an observation. Referencing to an element in a matrix is done using two subscript numbers; i, j, where i indicates the row number and j indicates the column number, e.g. $\mathbf{A}_{i,j}$. If only one subscript i is written for a matrix, this indicates the ith column. If nothing else is written, $|\mathbf{A}|$ indicates the number of observations/columns in the matrix.
- $\mathbf{A}^T$ is the transpose of $\mathbf{A}$
- $\mathbf{A}^{-1}$ is the inverse of $\mathbf{A}$

List of Abbreviations

2D Two-dimensional

3D Three-dimensional

CA Corrected Age

CP Cerebral Palsy

DoF Degree of Freedom

GM General Movement

FM Fidgety Movement

LM Levenberg-Marquardt

PCA Principal Component Analysis

WM Writhing Movement

Chapter 2

Clinical Aspects

This chapter introduces the clinical aspects of Cerebral Palsy (CP) and how this affects the life of people diagnosed with the disorder. Different methods for treatment are summarized as well. The chapter also considers early diagnosis of CP and introduces the so-called General Movements (GMs).

2.1 Cerebral Palsy

CP is the most common motor disability among children, affecting 2-2.5 out of 1000 infants [30]. It is caused by an injury of the fetal or infant brain and the physical impairment is in many cases accompanied by disturbances of cognition and perception [8]. During infancy, it can be difficult to observe symptoms of CP. The child is often diagnosed later in life, when the child is not able to meet certain motor milestones, such as crawling and walking [48]. Due to the difficulty in detecting CP, most children are not diagnosed until the age of 2 years [30]. When the child has been diagnosed with CP, there is no cure and the person will thus live with the disorder for the rest of his/her life. However, rehabilitation, surgery and medication can be used to reduce the effects. In addition to diagnosing a person with CP, the disorder can be classified based on the affected body parts and the severity. The following list summarizes the

different terms for the affected body parts, where -plegia and -paresis refers to paralysis and weakening, respectively:

- Mono-plegia/paresis: One body part is affected.
- Di-plegia/paresis: The disorder primarily affects the legs.
- Hemi-plegia/paresis: Arm and leg on one side is affected.
- Para-plegia/paresis: The complete lower body is affected.
- Tri-plegia/paresis: Three body parts are affected.
- Double Hemi-plegia/paresis: Both sides are affected, but one side is affected more than the other is.
- Tetra-plegia/paresis: Both arms and legs are affected, but one body part is less affected.
- Quadri-plegia/paresis: Both arms and legs are affected.
- Penta-plegia/paresis: Both arms and legs are affected as well as the head and neck.

In addition to classifying the disorder based on the affected body part, a scale known as Gross Motor Functioning Classification System is used. Here, the disorder is classified according to the person's ability to move with or without assistance:

- Class I: The person can walk without limitations, but has decreased balance and coordination.
- Class II: The person is able to walk up stairs with support for hands, but has troubles running and jumping.
- Class III: Walks with assistance from mobility devices, but might not be able to walk up stairs.
- Class IV: May not be able to walk, even with assistance and will most likely be needing a wheelchair for longer distances.
- Class V: The person is not able to move his/her body voluntarily and has problems with maintaining the head and neck upward.

Furthermore, CP can be classified based on the kind of motor disorder, which is closely related to the location of the brain injury. The different types of CP are summarized in Table 2.1.

Table 2.1: Description of the different types of CP

Type	Type of symptoms	Affected part of brain
Spastic	Muscles appear stiff	Motor Cortex
Dyskinetic	Involuntary movements	Basal Ganglia
Ataxic	Tremor-like motions	Cerebellum
Hypotonic	Muscles appear weak	Cerebellum
Mixed	Mix of above	Combination of above

2.1.1 Treatment

Once a person is diagnosed with CP, a number of methods exists that can help the person get better control over his/her movements. As mentioned, the disorder cannot be cured and the treatments often requires repeated actions throughout life.

- Physical Therapy: The goal here is to rehabilitate using exercises, in order to strengthen muscles and improve balance. The exercises can e.g. be based on using the muscles through participation in different kinds of sports, or it can be accomplished by doing a number of predefined exercises. Some studies focus on making the exercises more entertaining through the use of computer games and even make the intervention possible from home, while maintaining the possibility of individualizing the training [10].

- Botox: Lately, Botox injections has been used to make stiff muscles more relaxed. The treating effects of Botox last for periods of 3-6 months. Studies exist describing the positive effects of using Botox for treating CP, but there are uncertainties on how Botox effects the muscles in the long term [25] and there are no overall agreement whether or not Botox should be used for treating CP.

- Surgery: Some people tend to surgery where the muscles and tendons are manipulated directly. This can e.g. be operationally shortening or lengthening the muscle fibers, in order to tighten or relax the different limbs.

- Devices: Different devices can be used for assisting people with CP. The devices can e.g. be specially designed bikes, walking devices or wheelchairs, which helps the person move around more easily. Other devices focus on "removing" the involuntary movements, e.g. by the use of braces that fix the limbs in certain positions.

The listed techniques for treatment are only a few examples, but note that there is no solution that fits all. The solution should be found through communication between the person/family and the doctors/therapists. The severity of the disorder should also be considered, when a treatment is chosen.

2.2 Early Intervention

The above-mentioned treatment methods are often directed toward persons that are older because the disorder is often diagnosed when the person begins to walk. Various studies focus on early intervention of infants and how this affects the motor and cognitive development. However, the studies are often based on high-risk infants, whom might not have CP [29, 26]. Studies usually focus on different therapeutically techniques [72, 12] as well as nutritional interventions [88], and the effects of early intervention are not always conclusive. The reasons are often that the intervention is not conducted on infants with CP or because a true control group is not included. Most people will agree that it is simply not ethically correct to prevent intervention of infants that have CP. Extreme cases exist where infants were not stimulated during infancy and this led to decreased motor and cognitive development [29]. However, such cases are rare and the scientific extent is limited. Even though the effects of early intervention is not conclusive, this does not necessarily mean that early intervention should be discarded. Instead, focus on early diagnosis should be considered in order to conduct more controlled studies.

2.3 Early Diagnosis

In order to give an early diagnosis of CP, different approaches have been considered and reported in the literature. As CP is caused by an injury to the brain, one method is based on neuroimaging, i.e. imaging the brain in order to detect abnormalities [54]. These methods show promising results for predicting CP. However, as explained in [1], the different neuroimaging methods have some disadvantages, such as the time needed to do the imaging, exposure to radiation or operator dependencies. Other approaches focus on physiological or neurological measures, but these measures are less predictive of CP [54]. One method that has proven to be strong predictor for CP, is the general movement assessment method [18], which will be clarified in the following section.

2.3.1 General Movements

Besides the later symptoms of people with CP, studies have shown that the movement patterns of infants are influenced already in the months before and after birth [18]. These studies focus on so-called general movements, which are important in the development of the infants' motion control center. Furthermore, the general movements change behavior depending on the age of the infant [64]. The age of the infants is usually specified by Gestational Age (GA), Corrected Age (CA) or Chronological Age (ChA).

- GA is the age of the infant or fetus after the mother's last menstrual period (LMP). This age is usually used for infants born prematurely, before the term-age, which is 40 weeks GA. A preterm infant born 12 weeks before term will thus be 28 weeks old at birth.

$$\text{GA} = \text{Today} - \text{LMP} \tag{2.1}$$

- CA is the age of the infant, calculated from term. A preterm infant will thus have a negative CA. An infant born 12 weeks before term will be 0 weeks old 12 weeks after birth.

$$\text{CA} = \text{Today} - \text{Term} \tag{2.2}$$

- ChA is the age calculated from birth, i.e. the measure people usually use to describe their own age.

$$\text{ChA} = \text{Today} - \text{Birth} \tag{2.3}$$

As mentioned, the GMs are age dependent. GMs have been observed in 8-10 weeks old fetuses GA [15] and continues until the 12nd-20th week CA where the more intentional movements begin to dominate. The GMs are not always present and a number of parameters can affect the presence of the movements, e.g. if the infant is; using a pacifier, focused on activity around the infant, crying, playing with toys or sleeping[18]. The GMs can be divided into two kinds of movements, namely Writhing Movements (WMs) and Fidgety Movements (FMs). There might be some overlap between the two groups, but around 6-8 weeks CA, the movements change from WMs to FMs [64].

2.3.1.1 Writhing Movements

The normal WMs are categorized by smooth sequences of movements in arms, trunk and legs. The movements appear by waving through the body, e.g. by

starting in the arms, going through the trunk and the legs and returning to the trunk and the arms. The movements are random in the sense that the amplitude and speed of the movements varies a lot. The WMs also contain many rotations in the limbs, which enhance the smooth waving movements. For the WMs a number of abnormal movements are described in the literature [18], namely Poor Repertoire, Chaotic and Cramped Synchronized;

- Poor Repertoire: These movements are not very different from the normal WMs. However, the variability is reduced and in the obvious case, the infant is repeating the same monotonous movement with almost no change in speed or amplitude. The movements can also begin as normal movements but suddenly come to a halt instead of gradually continuing and fading out.
- Chaotic: These movements are somewhat similar to normal WMs, but the amplitude and speed is too large and the result is a very disorganized movement pattern.
- Cramped Synchronized: The movements consist of simultaneously contraction and relaxation of all limbs and the movements appear very rigid. The movements relates to a poor repertoire, in the sense that the movements of an infant with cramped synchronized movements is also classified as poor repertoire.

The age of the infant influences the WMs, as younger infants tend to have bigger amplitude/range in the motions [18] and this should be accounted for during the assessment of the movements.

2.3.1.2 Fidgety Movements

The FMs are described by less random movements, with smaller amplitude, compared to the WMs. These movements are usually very small movements in the body joints, resulting in a rocking or fidgeting behavior. It should be noted that other movements can occur as well such as kicking, swiping and wiggling due to excitement and these movements are considered natural, but are not related to FMs [18]. The abnormal FMs can be categorized as either abnormal or absent;

- Abnormal Fidgety Movements: In the case where the FMs are too big and chaotic the movements are described as abnormal. The movements

of an infant with abnormal FMs can be compared with the movements of a marionette. Compared to normal FMs the variation in the movements are reduced. However, an infant with abnormal movements can still turn out with normal motion development.

- Absent Fidgety Movements: In the case where no FMs are observed the abnormality is based on the lack of the normal FMs, which is very important in the development of the motor control skills. The lack of FMs might thus be used as an indicator for a motor dysfunction.

Studies have shown that the predictive power of assessing FMs is higher than assessing WMs [20, 18].

2.4 Concluding Remarks

CP is a motion disorder that affects both motor, cognitive and perceptual skills. The disorder is caused by damage to the infant brain, but the diagnosis is often given later in life. However, the disorder affects the movements of the person, already before birth. By observing and assessing the movements of infants, CP may thus be suspected at a much earlier age. Early identification of infants at risk of CP leads to the possibility of early intervention, which may improve the development of the infants' motor and cognitive skills. However, the early diagnosis is often based on qualitative measures. Due to this, a more objective solution is thus desired that is able to quantitatively measure the motion of infants.

Chapter 3

Measuring and Modeling Motion

In order to assess the movements of infants, it is necessary to measure signals related to motion. Different devices/systems already exist that are able to do this. The measurements can be based on internal body signals, i.e. signals in the body that causes the motion. These signals are called biosignals. Other devices measure the external motion, in the sense of measuring the displacement of different body parts. In this chapter, different systems for measuring motion are considered.

3.1 Biological

One way of measuring motion, is to measure the biosignals that cause the motion. The reason why a person moves is that the brain sends signals from the motor cortex, through the spinal cord and the nerve system, to the muscles that are activated in order to do some motor task. Due to this, motion can be measured, based on measuring the activity in the brain and the muscles. Using electroencephalography (EEG), electrical activity in the brain can be measured using small conductive sensors, attached to the subject's head. However, using EEG measurements, the data can include other signals not related to the

observed movement. Therefore, another approach is to measure the activity in the muscles. This can be done using electromyography (EMG), where sensors can be attached to the skin close to the location of the muscle. When a muscle is activated, this can be seen as changes in the EMG signal. For both EEG and EMG, the measurements can either be done invasively, where needles are in direct contact with the brain or muscles, or it can be done using surface electrodes. The latter approach is easier to conduct, but at the cost of a lower signal-to-noise ratio. However, even though both EMG and EEG are measuring (noisy versions of) signals controlling motion, this might not necessarily be enough to describe the complex movements of a person.

3.2 Non-optical

The non-optical systems use devices such as accelerometers, inertial measurement units or magnetic field sensors/emitters. These devices directly record the spatial positions (or derivatives of these) of a set of points. This is e.g. the concept behind the Nintendo Wii console [41]. The advantage of non-optical systems is that each sensor is often uniquely defined and no analysis is needed to discriminate these. Moreover, the position/velocity/acceleration is obtained directly from the sensors and no post processing is required. In addition, these systems do not have the drawback of overlapping limbs shadowing certain points of interest, as the sensors does not influence each other. The systems are usually used for problems, which require robust measurements or if the environment is not suited for optical motion tracking, e.g. when the tracking is done outside. Companies developing non-optical motion tracking systems are e.g. XSens[1] and Synertial[2].

3.3 Optical

In the case where some sort of camera is used the system is referred to as optical. Here, the motion induces changes in the recorded data. In order to model the motion, features should be tracked over time, which usually describe the motion of individual body parts. In the case of motion tracking, this can be done by use of feature extractors, which locate special features on the patient or by use of markers attached to the patient's body parts.

[1] www.xsens.com
[2] www.synertial.com

3.3.1 Marker-Based

When sensors are attached to the surface of the human, the motion tracking is referred to as marker-based. These markers can be either passive or active:

- Passive Markers: The markers are placed on the body surface and recognized by means of their appearance (such as color) or properties (such as retro-reflectivity). The problems with passive markers arise with the differentiation of multiple markers (marker swapping). This can be solved using unique appearances for the markers, e.g. based on using different colors.
- Active Markers: Active markers actively emit optical signals that can be used to track and identify the individual markers. This can e.g. be done using Light Emitting Diodes (LEDs), where the light is recognized and recorded by the optical system. The advantage of the active markers is that by controlling the LEDs, each LED can uniquely be identified and the problem of marker swapping is avoided.

Examples of marker-based systems are e.g. Vicon[3], Optitrack[4] and Qualisys[5]. The advantages of using markers, is that as the user of the system places the markers, the system is not model-specific. Marker-based motion tracking systems can thus be used for tracking arbitrary objects, but most of the systems focus on motion tracking of humans. However, some disadvantages of especially human motion tracking, is that the markers are placed on the surface of the human and the positions of the joints are often based on inter- or extrapolation [69].

3.3.2 Markerless

In the case where no markers are used, the system is called markerless. Here the motion tracking is based on identifying and tracking objects only using cameras. This is done using image analysis, computer vision and object recognition. The input to these methods can be Two-dimensional (2D) images originating from ordinary cameras or Three-dimensional (3D) volumes acquired with a 3D depth sensor. Some studies work with other modalities, such as thermal images in order to detect and track motion of humans [22]. One big advantage of a

[3] www.vicon.com
[4] www.optitrack.com
[5] www.qualisys.com

markerless system is that the system is contactless and will not influence the motion of the tracked person. A markerless system can e.g. be used for surveillance purposes. Examples of markerless systems are the tracking system from Organic Motion[6] or the Microsoft Kinect Sensor [7].

3.4 Modeling Humans

For some of the above-mentioned systems/devices for measuring motion, the output is a sparse set of spatial points. Even though these points alone are able to explain the human motion, the underlying skeleton is often modeled. The human skeleton consists of a fixed number of bones (typically 206 for adults and 270 for newborn infants). The bones shape the body and define the posture. The bones are connected by joints and these can move in a various number of ways, based on the type of joint. Due to this, motion tracking is usually modeled, by using an articulated model that represents the skeleton structure of a person. This articulated model can be represented as a graph or tree, where the joints/nodes connect the bones/edges and different transformations are related to the joints. The complexity of the articulated model depends on the desired level of detail. If a robotic arm is to be modeled, it might be enough to model this using 2-3 bones, but there are no limitations to the number of bones included in the model. However, modeling the full set of human bones is usually inappropriate and in most studies, it is sufficient to model the human skeleton using 10-20 bones. This is because many of the bones in the human body are fixed and moves only due to their connection to other bones. The number of modeled bones should of cause be chosen based on the purpose of the study and the desired level of detail. Existing studies model the motion of a hand, in which case the hand is modeled using 19 bones, describing the palm and fingers [38]. However, when the hand is not the primary structure, this is usual modeled using a single bone or as a part of the lower arm bone.

Once the desired number of bones is chosen, the next step is to model the movement of the bones. The straightforward approach would be to describe the body as a finite number of points/joints in space and their relative connections [24]. Modeling and tracking the motion of such a model is simply based on directly displacing the joint locations. Furthermore, physical parameters such as velocity and acceleration are easily calculated and constrained from frame to frame. However, even though this model is easily implemented, the model does not reflect the dependencies of different body parts. It is thus difficult to add relational constraints to the model.

[6] www.organicmotion.com
[7] dev.windows.com/en-us/kinect/

In most literature, a tree-based model is used instead, where a body part is chosen as the root node and the remaining body parts are either directly connected to the root node or indirectly connected through their parent body parts. Each node in the tree can thus be considered as a body joint. Furthermore, by representing the body in this way, the relative position of one joint with respect to its parent, can be represented by a simple translation and rotation. This makes it easy to implement dependencies on joints, as joints are naturally connected with their parent joints. If a person moves his/her fingertip, this should not affect other joints, but if he/she moves his/her shoulder, this will influence the position of the arm, hand and fingers, as they all inherit the change of the parent limb. The natural choice as the root is either the head [77, 76, 80] or the stomach/torso [81, 50, 91]. However, as seen in [21], other body parts can be chosen as well, which might be better suited in the specific case. In Figure 3.1 a typical skeleton structure is visualized with its associated tree representation in Figure 3.2, where the stomach is used as the root.

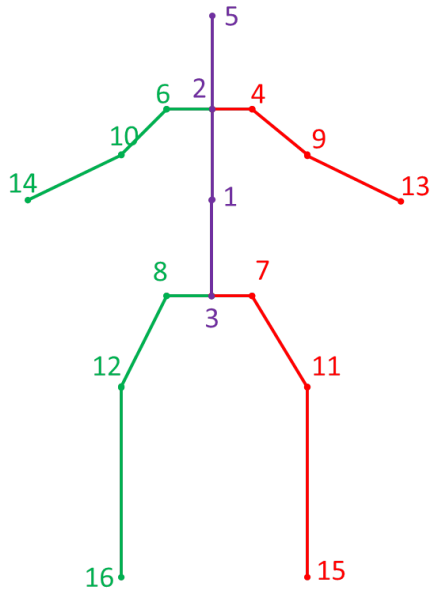


Figure 3.1: The articulated model visualized in a skeleton structure. The skeleton structure assembles that of the human figure (front towards the reader).

Moreover, anatomical constraints can easily be incorporated in the model, such as limitations to the limbs' lengths (scaling) or orientation (rotation). The length/size constraints can be based on knowledge about the human body, which has been studied for centuries (c.f. The Vitruvian Man and the studies by the

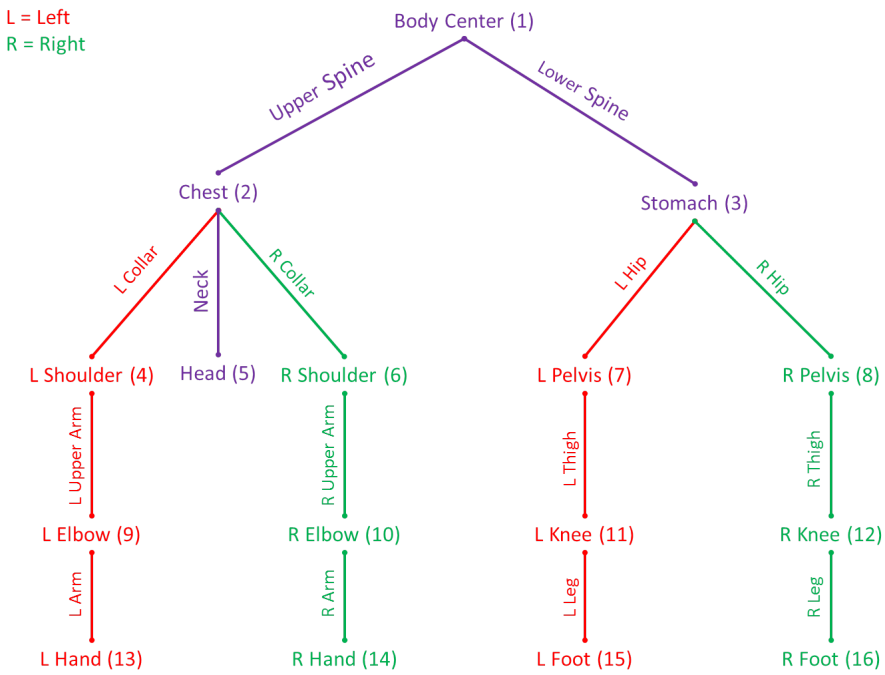


Figure 3.2: The articulated model visualized in a tree structure. The tree structure gives a good overview of the different joints and their dependencies.

Roman architect Marcus Vitruvius Pollio). From this, proportional constraints on the human body can be defined e.g. based on the total height of the person [55]. Other studies fix the length of the body parts [28]. This fixed length can be set either before the motion tracking or during the motion tracking, such as in [82], where a predefined pose is acted by the subject. The T-pose is often used, i.e. a pose in upright position with the legs joined and the arms going out horizontally. From this, simple heuristics can be used to automatically estimate the body dimensions. Regarding the rotational limitations, these can also be determined anatomically, based on the different joint types connecting the respective limbs. In physiology, six different joint types are used to describe the possible movements of joints [71]. However, when modeling the human, three joints are often used, discriminated by the Degrees of Freedom (DoFs), i.e. the minimum number of parameters needed to describe their motion [87, 62, 11]:

- Hinge or Pivot: Has 1 DoF and can thus rotate around one single axis. The ankle is an example of such a joint and more specifically a hinge joint.

- Saddle or Ellipsoidal: Has 2 DoFs and can thus rotate around two axes. Body joints that are ellipsoidal joints are e.g. the wrist.

- Ball-and-Socket: Has 3 DoFs and can thus rotate around three axes. Body joints that are ball-and-socket joints are e.g. hip and shoulder.

In the following section, the mathematical representation of a joint is considered.

3.5 Forward and Inverse Kinematics

In order to mathematically describe the articulated model, the concepts of forward and inverse kinematics are introduced. Due to the articulated structure of the human model, the rotations of one limb will affect the child-limbs and the position of a joint can be written as a combination of transformations. The joint is also called the end-effector, i.e. the point that is affected in the end of the chain.

$$\begin{aligned} \mathbf{p}_g &= \mathbf{T}_0\mathbf{T}_1 \dots \mathbf{T}_k\mathbf{p}_l && (3.1) \\ &= \mathrm{FK}(\theta) && (3.2) \end{aligned}$$

Here, $\mathbf{p}_l$ and $\mathbf{p}_g$ are respectively the local and global position of the point $\mathbf{p}$ and $\mathbf{T}_i$ is the local transformation matrix for the i'th parent limb of $\mathbf{p}$, ordered from the root. The transformation matrix $\mathbf{T}_0$ is the global transformation for the root and this is common for all other end-effectors. For simplicity, the chain of transformations is combined into a forward kinematics function FK, which transforms the local point based on a parameter vector θ. It should be noted that the chain of transformations is different for every end-effector and there is theoretically a forward kinematic function for every end-effector.

In many cases, the parameter vector θ is unknown, but the spatial location of the end-effector is known. This is often the case, when using optical marker-based motion tracking systems. The goal is thus to find the set of parameters that minimizes the distance between the known joint location $\bar{\mathbf{p}}$ and the model's end-effector position $\mathrm{FK}(\theta)$. This concept is known as inverse kinematics and is usually solved using optimization, where the solution in Equation 3.3 is sought.

$$\theta^* = \arg\min_{\theta} ||\bar{\mathbf{p}} - \mathrm{FK}(\mathbf{p}_l|\theta)|| \tag{3.3}$$

$$= \mathrm{IK}(\bar{\mathbf{p}}) \tag{3.4}$$

Again, for simplicity, the equation is reduced to a single function IK (See Equation 3.4), which depends on the respective body part. Often multiple end-effectors are considered simultaneously, leading to:

$$\theta^* = \arg\min_{\theta} \sum_i ||\bar{\mathbf{p}}_i - \mathrm{FK}(\mathbf{p}_i|\theta)|| \tag{3.5}$$

There is no closed-form solution to the inverse kinematics problem and the solution is often found using an iterative solver based on a linesearch or steepest descent optimization. However, it should be noted that the solution is rarely unique and different configurations can lead to the same minimal cost. This problem is known as motor redundancy.

3.6 Rotation Parametrizations

To mathematically describe the rotation of the joints, a number of representations are used in the literature. The different representations will be summarized below, accompanied by the formulas for calculating the rotation matrices.

3.6.1 Euler Angles

A natural way of describing rotations is to define the rotation around the X-, Y- and Z-axes, given the angles of rotation for the three axes. Using the definitions of rotations in a right handed 2D coordinate system, the rotation matrices for

the three axes can be defined as:

$$\mathbf{R}_X(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{pmatrix} \tag{3.6}$$

$$\mathbf{R}_Y(\beta) = \begin{pmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{pmatrix} \tag{3.7}$$

$$\mathbf{R}_Z(\gamma) = \begin{pmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{3.8}$$

Here α, β and γ are the angle of rotation for the three axes. The different rotation matrices can be combined by multiplication, in order to describe multiple subsequent rotations. As matrix multiplication is not commutative, the order of the multiplications should be considered. Furthermore, the Euler angle representation suffers from the so-called Gimbal lock, which occurs due to singularities in the Euler angle representation. It is thus possible to end up with configurations where changing one of the angles will not result in a change in the rotation matrix. It might also happen that two rotation matrices, which might look similar, are constructed from two completely different Euler angle configurations. This can especially lead to problems, when the rotation term needs to be inter-/extrapolated, which is often the case in motion tracking and computer graphics. However, the problem can be avoided, if the motion tracking is well constrained.

It should be noted, that the Euler angle representation is generalized by the axis-angle representation, where the representation consists of both an angle of rotation as well as the axis of rotation. Given the angle of rotation α as well as the axis of rotation $\mathbf{v}$, the rotation matrix can be generated from the Rodrigues formula:

$$\mathbf{R}(\alpha, \mathbf{v}) = \mathbf{I}_3 + \sin(\alpha)\mathbf{V} + (1 - \cos(\alpha))\mathbf{V}^2 \tag{3.9}$$

$$\mathbf{V} = \begin{pmatrix} 0 & -\mathbf{v}_z & \mathbf{v}_y \\ \mathbf{v}_z & 0 & -\mathbf{v}_x \\ -\mathbf{v}_y & \mathbf{v}_x & 0 \end{pmatrix} \tag{3.10}$$

It should be noted that one angle and three parameters for the direction vector is used, i.e. four parameters in total. However, as the vector is only used for direction, the normalized vector can be scaled with the angle and the rotation can be represented by three parameters. Using Equation 3.9 with the three canonical basis vectors leads to the rotation matrices for the usual Euler angle representations.

3.6.2 Quaternions

Another representation for rotations is the unit length quaternions, which can be seen as a four-dimensional vector. Based on the elements of the vector the quaternion encodes an axis-angle representation with a rotation α and an axis of rotation $\mathbf{v}$.

$$\mathbf{q} = \begin{pmatrix} \mathbf{q}_x \\ \mathbf{q}_y \\ \mathbf{q}_z \\ \mathbf{q}_w \end{pmatrix} \tag{3.11}$$

$$\alpha = 2\cos^{-1}(\mathbf{q}_w) \tag{3.12}$$

$$\mathbf{v} = \frac{\begin{pmatrix} \mathbf{q}_x \\ \mathbf{q}_y \\ \mathbf{q}_z \end{pmatrix}}{\sin(\cos^{-1}(\mathbf{q}_w))} \tag{3.13}$$

The inverse conversion from axis-angle representation to quaternion representation is easily derived from the inversion of the these equations.

Given the unit quaternion, the rotation matrix can be generated using the following formula:

$$\mathbf{R} = \begin{pmatrix} 1 - 2\mathbf{q}_y^2 - 2\mathbf{q}_z^2 & 2\mathbf{q}_x\mathbf{q}_y + 2\mathbf{q}_w + \mathbf{q}_z & 2\mathbf{q}_x\mathbf{q}_z - 2\mathbf{q}_w\mathbf{q}_y \\ 2\mathbf{q}_x\mathbf{q}_y - 2\mathbf{q}_w\mathbf{q}_z & 1 - 2\mathbf{q}_x^2 - 2\mathbf{q}_z^2 & 2\mathbf{q}_y\mathbf{q}_z + 2\mathbf{q}_w\mathbf{q}_x \\ 2\mathbf{q}_x\mathbf{q}_z + 2\mathbf{q}_w\mathbf{q}_y & 2\mathbf{q}_y\mathbf{q}_z - 2\mathbf{q}_w\mathbf{q}_x & 1 - 2\mathbf{q}_x^2 - 2\mathbf{q}_y^2 \end{pmatrix} \tag{3.14}$$

The reason for adding the additional DoF, compared to the Euler angle representation, is that the representation is free from the Gimbal lock problem and changing the quaternion slightly, changes the rotation matrix similarly. This makes interpolation possible and generates smooth motion when used in computer graphics. However, one problem with the quaternion representation is that the quaternion needs to be unit length and thus the quaternion should be normalized before it is used for rotation. As in the axis-angle representation, the rotation can be represented by three parameters instead of four parameters. This is under the assumption that the quaternion is indeed normalized to unit-length.

3.7 Concluding Remarks

Different systems for measuring motion can be used. As mentioned in Section 2.3.1, the general movements are affected by external stimuli and thus the measurements should be acquired without effecting the infant. Furthermore, a future goal with this study is to develop a tool that can assist therapists in assessing the movements. Technical calibration procedures and preparation should thus be limited. Based on these desired properties a markerless system is the best choice, as the measurements can be done without the need for attaching sensors on the infant's body.

Chapter 4

Existing Work

As mentioned, the Kinect sensor has a markerless motion tracking system build in and its motion tracking system is a relative new technique. However, motion tracking using optical sensors has been studied for several decades, using different modalities. In the following, existing work on optical motion tracking is described, with focus on motion tracking using RGB-D data, i.e. data containing both color and depth/Three-dimensional (3D) information. Previous work on motion tracking of infants is considered, with emphasis on detecting movements related to Cerebral Palsy (CP).

4.1 Human Detection

Before the pose estimation and motion tracking can be applied, the goal is to detect humans in the data. Several approaches to this problem exist, where the detection is based on assumptions regarding the human body [19]. This can e.g. be based on finding regions of movements; based on optical flow, background subtraction or feature extraction, e.g. in the sense of gradient based features. The shape of the regions can also be used to detect possible human objects, e.g. using the assumption that the human is posing in a certain position. In studies on pedestrian detection, it can usually be assumed that the camera is aligned

such that the pedestrians are horizontal objects in the data. If training data are available, this can be utilized to extract features describing the appearance of humans and a classifier can be trained to detect these human-specific features. In [57], the authors describe a general framework for learning appearance features of any object, given enough training data. The authors demonstrate several applications, including detection of pedestrians. This approach is also the foundation of the well-known Viola Jones face detection algorithm [86]. As the method can be extended to detect any kind of objects, the methods can be used to identify individual body parts in the data [46]. A similar concept is the basis of the pose estimation algorithm in the Microsoft Kinect Sensor [75]. Here, a huge database of depth images of human figures are used for training. The features are based on relative differences between neighboring regions in the depth data. Other features for human appearance has been used as well, including the popular gradient-based feature known as HoG features [13]. A new popular approach is to use deep neural networks in image analysis and computer vision. This has been studied within human pose estimation [83], where both the pose model as well as the features are learned automatically.

4.2 Estimating Poses

Once the human has been detected and localized in the data, the next step is to estimate the pose. In some studies, this step is solved simultaneously with the detection of the human, as the human detection is based on finding the different body parts. Connecting the identified body parts results in an estimate of the respective skeleton [75, 46]. However, such methods does not necessarily incorporate information/assumptions regarding the anatomical structure of the human body. In [61, 7] it is assumed that the human body can be modeled as a fixed number of limbs (arms, legs and head), which is usually the case. By representing the observed (and segmented human) data as a graph, the outer body parts can be found as points farthest away from the center of the body. This gives a number of candidates for the extremity points. Here, the prior knowledge regarding the orientation of the camera or human can be used, to further detail the anatomical meaning. From this, a rough estimate of the human skeleton can be found and used for further refinement. The method has some drawbacks when body parts overlap or are close to each other, but the method is good for automatically estimating the pose of a human person. In [70] the method is improved, where optical flow is used to distinguish between overlapping body parts.

Other approaches focus on defining a full model of the human surface that can be changed using different orientation and size parameters. The model can e.g.

be explained in a part-based manner, where every body part is represented by a set of parameters. The choice of structures representing the body parts is often task-specific, but popular structures are cylinders, ellipsoids and spheres [53, 58, 74]. However, other structures are used as well throughout the literature, such as stickman models [65, 43] or Gaussian structures [79]. The model can include both shape and color, where the color model can be adapted to the specific data [23, 79]. Models with increased detail are also used in model-based pose estimation, where a mesh represents the surface of the tracked object or human [73, 93, 7]. Similar to the part-based model, a (usually higher) number of parameters represents the shape and posture of the model and the goal is to find the parameters that minimize the distance between the mesh and the observed data. In the case where the observed data are 3D point clouds, the metric can simply be based on the distance between the model surface and the point cloud. However, in studies where the data originate from depth and/or color images, it might be beneficial to project the model to these spaces by synthesis [53, 89]. This can e.g. be used to reconstruct the 3D pose of a human from a single color image. In the case where depth images are synthesized, the metric is usually based on differences between the true depth image and the synthesized depth image. Image synthesis is often used in motion tracking, but requires a good model of the possible poses. A closely related approach is to use machine learning, where several known poses and their respective Two-dimensional (2D) projections are used to train a pose classifier [5, 58]. The 3D pose in a new 2D image can thus be estimated based on the learned correlation between 2D images and 3D poses. One problem with this kind of tracking is that the classifier restricts the possible pose-candidates and poses not seen before are difficult to classify. Furthermore, these methods often require huge amounts of training data, which are not always available.

4.3 Filtering and Corrections

Until now, the motion tracking has been explained as the goal of estimating the pose of humans in single frames. The motion tracking is simply obtained by combining the poses between successive frames. Based on the robustness of the method for estimating the pose, it might be beneficial to incorporate filtering between the frames. First, the estimation can utilize previously obtained results of the pose and secondly, the filtering can ensure that wrong pose estimations are detected and corrected. In the case of real time motion tracking, the filtering is based on previous frames. The filtering can simply be linear combination of the current frame and the previous frames. However, more complex techniques exist that incorporate uncertainties on the estimated parameters, such as particle- and Kalmann filters [40, 42]. In the case of offline motion tracking, the filtering

approach can still be used. However, the estimated poses from future frames should be included as well. In addition to the filtering approach, the tracking results can also be corrected by physical properties e.g. by penalizing high velocities or intersecting body parts. If the motion tracking is model-based, the individual joints are often constrained by the range of rotation.

4.4 Motion Recognition

Once the motion has been obtained from the motion tracking system, the next and usually final step is to classify the activity of the human(s) being tracked. In order to do so, a number of features must be extracted from the motion tracking results. Popular features are based on distances, velocities, accelerations, frequencies and orientations [52, 56, 90, 49, 47]. In [52], the authors recognize actions such as standing, bending, walking and sitting. The features are based on relative angles between body parts as well as body part specific velocities and accelerations. The classification is based on relative simple heuristics of the different actions. Different sports actions are recognized in [56], such as punching, golf swing, throwing and kicking. Here, the classification is based on training a classifier on annotated sequences of the the different actions. Other methods rely on less supervised classification, such as clustering a sequence of frames, consisting of multiple successive actions [92]. The studies summarized here consider rather rough actions. This does not necessarily mean that the actions are classified easily, but the movements are often easily recognized by humans. Studies focusing on more complex/detailed actions are also considered in the literature, e.g. in the sense of recognizing sign language [49, 47] or for detecting symptoms of Parkinson disease [14].

4.5 Infant Tracking and Prediction

Even though motion tracking is a field that has been studied for many years, the work usually focuses on motion tracking of adults, while motion tracking of infants is sparsely studied. However, some studies do exist within this field and in particular with the goal of classifying and diagnosing motion disorders such as CP. In [45], the authors use an optical motion system, where reflective markers are attached to the infant's limbs and multiple infrared cameras are used to reconstruct the 3D locations of the markers in space. From this, a set of parameters is extracted and used for early detection of spasticity due to CP. The infants included in the study were too young to have Fidgety Movements (FMs), but the authors were able to correctly classify healthy and affected

infants with an accuracy of 73%. Reflective markers are also used in [32], where the markers are tracked in 2D using image analysis. The temporal tracking of the markers are used for estimating velocity and acceleration features, which are then used for classifying the movements. In [9, 66] a non-optical approach is used to extract motion parameters, where sensors are attached to the infants' wrists, ankles, chest and head. The sensors measure position and orientation and generates time series of these. For each frame a set of features are generated based on distances between points, periodicity, deviation from moving average, etc. Using machine learning, the features are used to train a classifier for detecting normal or abnormal movements. Similar features are used in [35, 36, 34, 59], where magnetic sensors are attached to the infants' limbs and used for measuring the spatial position over time. The authors describe features for segmenting and describing the variation of spontaneous movements in infants. The features are later used for diagnosing infants based on their movements. The above-mentioned studies focus on marker-based techniques but a few existing studies base the analysis on markerless approaches, using ordinary video recordings of infants. In [33], the authors propose a new optical flow-based method for quantifying the motion of infants with neonatal seizures. The study is based on an overall quantitative measure of movement between successive frames. An optical flow-based approach is also used in [78], where the authors use color images as input to an optical flow-algorithm, in order to track the position of the infants' arms and legs. Here, the authors manually initialize the position of the different body parts and adjust the positions during the tracking, in order to improve robustness of the method. The same group has previously looked at other motion features extracted from video recordings of infants. In [4] the authors present the General Movement Toolbox, which is a tool for visualizing and assessing movements in video recordings. The authors generate so-called motiongrams, which captures temporal changes in images. From the motiongrams, different quantitative features are extracted and used for classifying abnormal movements in infants. The method was later used for further validation, showing promising results [2, 3]. In [67], the authors segment the infant's different body parts, leading to the possibility of tracking the different limbs in 2D without the need of markers/sensors attached to the infant.

4.6 Concluding Remarks

Studies on infant motion tracking already exist. In these studies, the authors base their analysis on data recorded with the goal of diagnosing motion disorders such as CP. Due to this, the developmental outcomes of the infants are often known and used for supervised classification. The studies are thus adapted for classifying specific movements. However, only few of the studies consider

modeling the body of the infant using general motion tracking techniques. Doing so might lead to a more general description of the infants' movements and can thus be used as a general tool for movement assessment.

Based on existing studies, it is obvious that many different techniques can be used for markerless motion tracking. However, not all of them are applicable for infants, as some of them are based on prior knowledge of adult appearance and motion. Furthermore, keeping in mind that the study should focus on assisting clinicians, it is important that the required preparation/calibration is kept simple. Including the fact that the goal is to monitor the infant's motor development, a favorable property could be to do the tracking at the infant's home, thus putting further emphasis on simplicity. In this study, we choose to use a depth sensor for doing the data recording. The sensor is able to capture both color and depth information, thus giving additional information of the shape of the infant. Furthermore, existing work have shown that this kind of data is able to do impressive motion tracking of adults, without the need for calibrating the system.

Chapter 5

Data Acquisition

The data used in this project is based on image sequences of moving infants. The acquisition is done using the Microsft Kinect Sensor which will be explained in this chapter. In addition, the setup for recording the data is described. Furthermore, the chapter describes different approaches for background removal in order to segment the infant. It should be noted that as the data is recorded simultaneously with the study, the true outcome of the infants are unknown. Some of the infants are born preterm, but so far none of them have been diagnosed with Cerebral Palsy (CP).

5.1 RGB-D Sensors

This study focuses on motion tracking using RGB-D sensors, i.e. cameras that capture both color (RGB) and depth (D) information (See Figure 5.1). As the depth image explains the distance from the camera to the object, every pixel in the depth image corresponds to a point in Three-dimensional (3D) space/world-space and the transformation from depth-space to world-space is often known and even invertible. The 3D estimation together with their low cost is the primary reason why these sensors have become so popular.

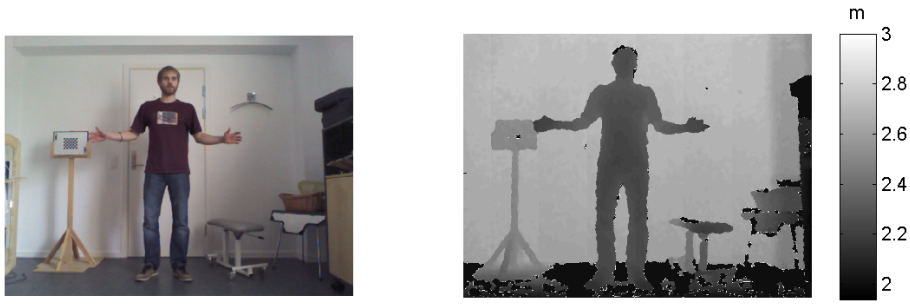


Figure 5.1: The data obtained from a RGB-D sensor are illustrated. Left: The color-data from the sensor is visualized. Right: The depth data is visualized as a gray-scale image. Dark regions represents objects close to the sensor, while bright regions are further away. The black regions cannot be estimated and have the value zero.

The technique to estimate the depth is usually based on one of the following methods:

- Stereo Vision: As the name indicates, this reconstruction technique is based on observing objects using two (or more) cameras. By observing an object from multiple viewpoints, the 3D geometry can be reconstructed using simple trigonometry. However, the difficult and time-consuming task of stereo vision is usually to find correspondences between the projected images. In the uncontrolled environment, this can be done using different feature detection approaches. Another approach is to use structured light, where the features are generated from a projector. Using a projector, a known light pattern can be projected onto the scene/object. A camera captures the reflection of the projected light and correspondences between the projected and captured light is found. This is under the assumption that the camera can see the projected light (light in the invisible spectra is usually used).
- Time of Flight: By projecting light out in a scene and measuring the time it takes the light to be reflected and return to the sensor, the distance can be measured. As the distance estimation is based on simple calculation of time-differences, this method can usually be done much faster than the stereo vision approach and with higher accuracy in the depth estimation. This technique is used in the Microsoft's second version of the Kinect sensor.
- Depth from (De-)Focus: A less used approach is depth from focus, where the depth is estimated from images captured with an ordinary color cam-

era. Due to the camera's internal properties, objects will appear blurred, when they are out of the focus range. The focus range can be estimated from the focal length, which should be known beforehand. By detecting when objects are in focus in images, it is possible to estimate the depth information, based on multiple frames, where different focus ranges are used. As mentioned, this technique is not used very often, but is mostly considered in scientific experiments.

5.1.1 Microsoft Kinect

In this study, the Microsoft Kinect Sensor is used (see Figure 5.2). The sensor came with the Microsoft XBox console in 2010 and in 2012 a Microsoft Windows version of the sensor was introduced which made it possible to (legally [31]) connect the sensor to a PC and process the data from the sensor. The depth estimation is based on an infrared structured light pattern projected by a Class I laser. This means that the amount of exposure never exceeds the limit acceptable for looking at the laser with the naked eye. This also means that the depth estimation does not work well outdoors, as the projected pattern will be saturated by the sunlight. The color and depth images are recorded with a sampling rate of 30 frames per second. It should be noted that as with any other measuring device, the data contains noise, which might have an impact on the further analysis. This is also the case with the Kinect sensor. A number of existing studies measure the precision and accuracy of the depth estimation [39, 44, 6]. The studies show that the noise increases exponentially with the distance between the sensor and the object. However, this study focuses on distances close to one meter and the depth resolution can be expected to be 2-3 mm.

In addition to the release of the low-cost RGB-D sensor in 2010, a major advantage of the sensor is its fast and robust motion tracking system [75]. The system is based on recognizing human body parts in the depth data, where the sensor has been trained to learn the appearance of these body parts. The body parts are connected based on the known skeleton structure and the detected skeleton is illustrated on top of the respective color-image in Figure 5.3.

However, as the motion tracking is based on learned depth appearance of adults, the system is not able to track the body of an infant. The sensor is only able to distinguish between adult body parts and background. In some cases, the system can create unrealistic poses, if pixels are wrongly classified, as illustrated in Figure 5.4. However, this usually occurs in scenes containing noisy regions in the sense of occlusions, multiple similar foreground regions, etc.

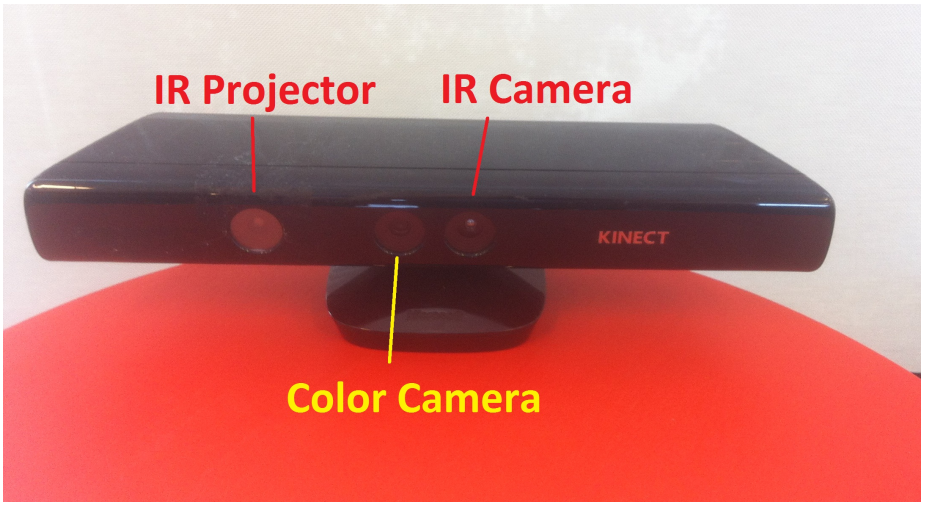


Figure 5.2: The depth sensor used in this study. The sensor consists of a color camera as well as a depth sensor, comprised of an infrared projector and an infrared camera.

In 2014, the second version of the Kinect (Kinect v2) was introduced together with the release of the Microsoft XBox One. The new sensor improves both color and depth resolution and the depth estimation is based on the time-of-flight technique. However, the improvements require a higher transfer rate between the sensor and the PC in the sense of a USB 3.0 connection. Furthermore, the second version requires that the PC have at least Microsoft Windows 8 installed. Due to these hardware/software limitations and because of the time of the release, Kinect v2 was discarded for recording of additional data. However, as the study is not hardware specific, the algorithms and methods explained in this work, can directly be applied to data from the improved Kinect v2 or any other RGB-D or 3D sensor. Both Kinect v1 and Kinect v2 contains multiple microphones, that can be used for sound recording and voice activation, but this is not used in the study and will be ignored.

5.2 Setup

As the goal is to record the infants' movements and especially the movements of the limbs, the position of the sensor with respect to the infant is important. The sensor is positioned right above the infant, as illustrated in Figure 5.5 and the infant is positioned in supine position, i.e. on its back. Initially a blue

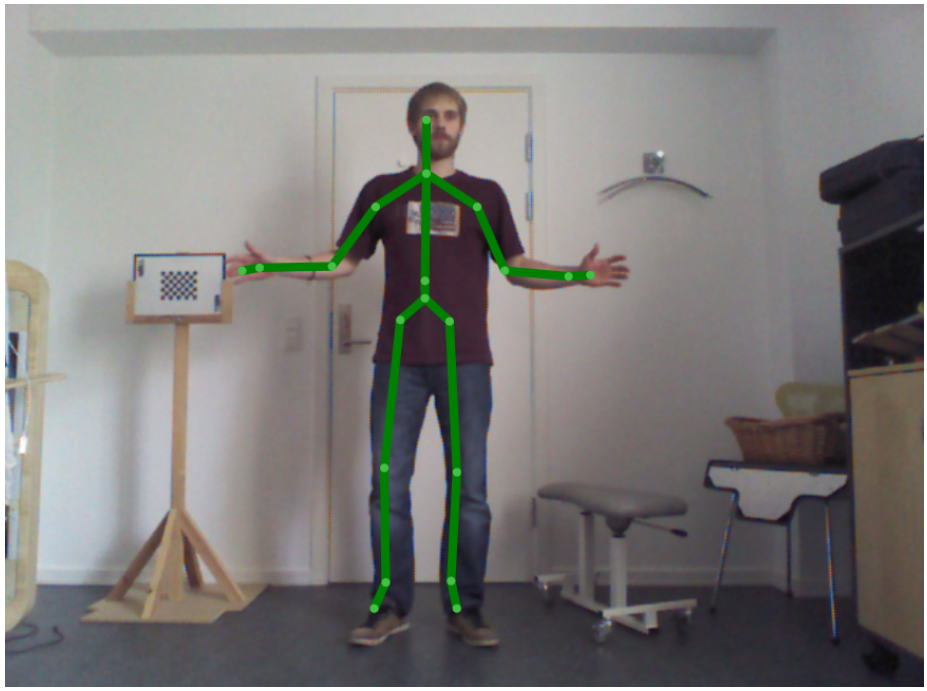


Figure 5.3: Based on the depth-based body part recognition approach, the Kinect is able to identify and connect the different body parts. The green lines represent the skeleton bones and the points represent joint-positions.

colored bodystocking simplified the task of automatically identifying the infant's torso and pelvis (see Figure 5.6). However, in later recordings the bodystocking was not used, as the later analysis only relied on the depth data. Some recordings were done at the Helene Elsass Center while others were done at the infant's home. In order to be able to capture the General Movements (GMs), we ensured that the infant was awake and in a good mood, according to the clinical prerequisites for observing GMs. Furthermore, we described the procedure to the parents beforehand and the parents could at any point choose to stop the recording. In addition, the parents signed a consent form, describing that we were allowed to use the data for research purposes and that the parents had been informed about the procedure.

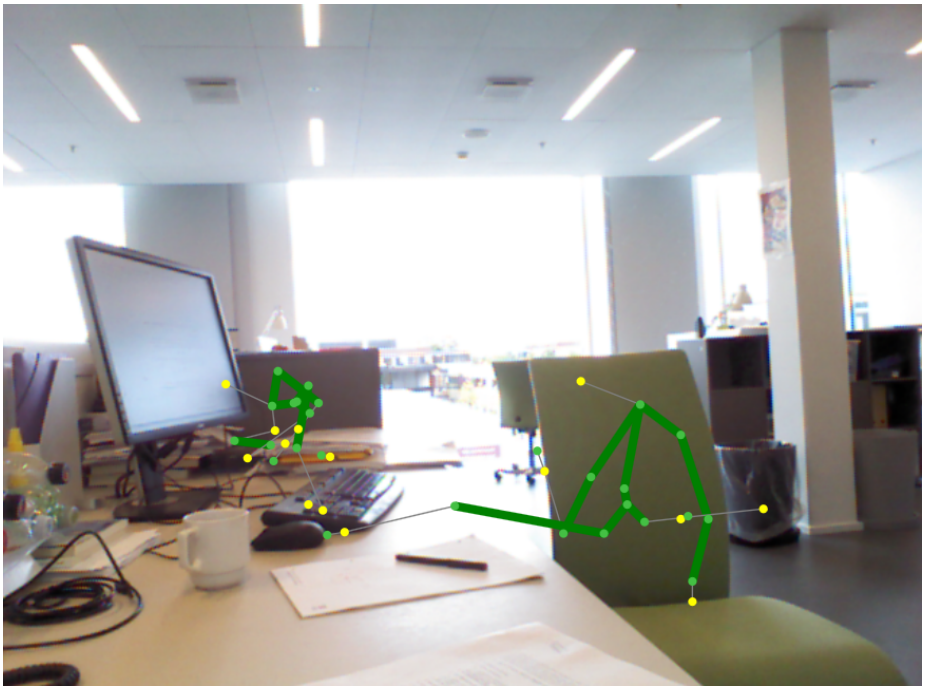


Figure 5.4: In noisy scenes, the Kinect might wrongly classify regions belonging to different body parts and thus an unrealistic skeleton pose is obtained.

5.3 Data Preprocessing

As explained earlier, the data from the Kinect are both color and depth data, as seen in Figure 5.7. The infant is lying in the center of the images, but the recordings also include the surroundings, such as the mattress, the floor, etc. Before the data can be used for motion tracking, a preprocessing step is applied. This step includes background removal in order to segment the infant from the rest of the scene. A number of methods can be used to do this background removal and this will be explained in the following sections.

5.3.1 Color Analysis

In some of the data, the infant is wearing a blue-colored bodystocking. By use of methods from image analysis, the bodystocking can be segmented and used for automatically locating the infant in the data. The choice of color for

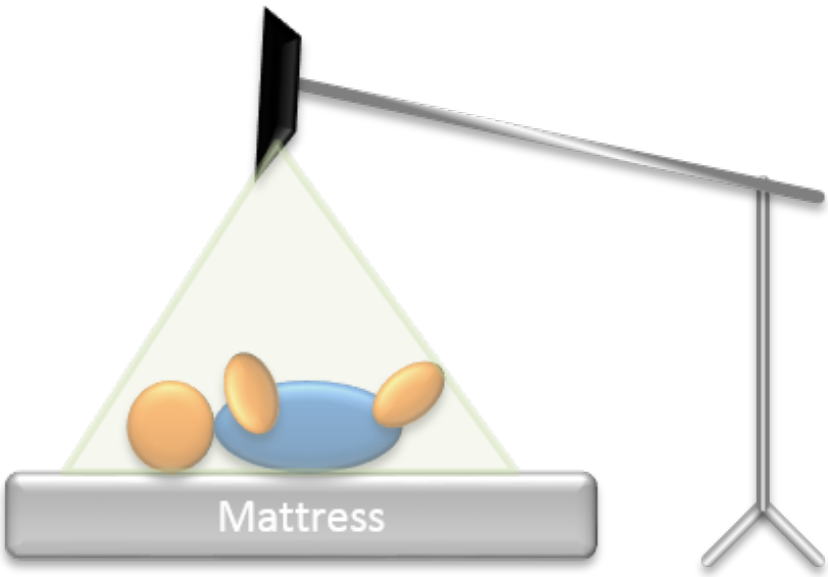


Figure 5.5: A simple illustration of the setup used during the recording of the infant's movements.

the bodystocking is based on the color being different to that of human skin (especially Caucasian skin). The background color is usually green or blue when doing chroma keying, which is e.g. used in movies and sometimes in weather forecasting in TV. The segmentation is done using simple thresholding for different color-channels, i.e. for every pixel in the image, the pixel is assigned a binary label, whether or not the pixel looks like a bodystocking pixel. In order to model the appearance of the bodystocking, a number of training-images is used to estimate the distribution of different color channels. In this work, the segmentation is done based on the RGB- and HSV-colorspace. Based on the learned distribution of the color, the Mahalanobis distance is calculated. Pixels are classified as bodystocking, if the distance is within the 99% percentile. More advanced segmentation strategies can be used, e.g. using Markov Random Fields [37], in order to enforce a neighborhood consistency. However, the simple threshold approach works well for the used data, as can be seen in Figure 5.8, where the bodystocking is segmented using the above-mentioned approach.

Figure 5.6: Some infants wore a blue-colored short-sleeved bodystocking to simplify the analysis.

Figure 5.7: An example of the data used in this study.

5.3.2 Background Removal

The recorded data contains both data from the infant (foreground) as well as the surroundings (background). In order to analyze the movement of the infant,

Figure 5.8: Using the color-information, the blue bodystocking can be segmented in the image and used for prior knowledge regarding the position of the infant.

the background information should be removed. This is a common task in many computer vision problems, where a filtering is applied to the input data in order to remove the unwanted data.

5.3.2.1 Background Modeling

If the background is known beforehand or if a number of background frames are available, this can be used to generate a background model. The background is usually modeled for each pixel and can e.g. be a simple scalar comparison. More complex methods can be used as well, which incorporates uncertainties on the background and/or measurements. In the simplest case, every pixel can be compared to a global parameter τ as in Equation 5.1.

$$\mathbf{L}_{i,j} = \begin{cases} 1 & \text{if } \mathrm{D}(\mathbf{Im}_{i,j}) \leq \tau \\ 0 & \text{if } \mathrm{D}(\mathbf{Im}_{i,j}) > \tau \end{cases} \tag{5.1}$$

Here, D is a function that weights the pixel-value based on the segmentation model. It might be that $\mathrm{D}(\mathbf{Im}_{i,j}) = \mathbf{Im}_{i,j}$, but in the case of multi-channel images it might be necessary to combine the intensities using linear combinations of the channel-intensities. Often, it is too simple to segment using a single threshold value and it might be that a pixel-dependent threshold is required as seen in Equation 5.2.

$$\mathbf{L}_{i,j} = \begin{cases} 1 & \text{if } \mathrm{D}(\mathbf{Im}_{i,j}) \leq \mathbf{T}_{i,j} \\ 0 & \text{if } \mathrm{D}(\mathbf{Im}_{i,j}) > \mathbf{T}_{i,j} \end{cases} \tag{5.2}$$

A generalization of the two equations are written in Equation 5.3, where the segmentation can include more complex models, e.g. using multivariate classification or Gaussian mixture models [60].

$$\mathbf{L}_{i,j} = \begin{cases} 1 & \text{if } \mathrm{B}(\mathbf{Im}|i,j) \text{ is true} \\ 0 & \text{if } \mathrm{B}(\mathbf{Im}|i,j) \text{ is false} \end{cases} \tag{5.3}$$

5.3.2.2 Plane Fitting

In our data, the infant is positioned on a flat surface, usually a mattress or a blanket. This flat surface can be considered as lower a boundary for the scene, i.e. objects "below" this boundary is removed. This process is done using a simple plane fitting approach to the mattress points, based on Principal Component Analysis (PCA). Given a set of 3D points $\mathbf{Q} \in \mathbb{R}^{3\times n}$, a plane can be fitted to the data using the following approach.

1. Normalize $\mathbf{Q}$ with respect to translation:

$$\mu_{\mathbf{Q}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{Q}_i \tag{5.4}$$

$$\hat{\mathbf{Q}}_i = \mathbf{Q}_i - \mu_{\mathbf{Q}}, i = 1 \ldots n \tag{5.5}$$

2. Calculate the covariance matrix $\mathbf{\Sigma} \in \mathbb{R}^{3\times 3}$:

$$\mathbf{\Sigma}_{\hat{\mathbf{Q}}} = \hat{\mathbf{Q}} \cdot \hat{\mathbf{Q}}^T \tag{5.6}$$

3. Decompose the covariance matrix $\mathbf{\Sigma}$ into its Eigenvalues and Eigenvectors:

$$\mathbf{\Sigma} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1} \tag{5.7}$$

   The matrices $\mathbf{V}$ and $\mathbf{D}$ will then contain the Eigenvectors and Eigenvalues, respectively. The ith column of $\mathbf{V}$ is the Eigenvector associated with the ith diagonal element in $\mathbf{D}$. Alternatively to the Eigendecomposition, Singular Value Decomposition can be used directly on $\hat{\mathbf{Q}}$, where the singular values and right-singular vectors correspond to the Eigenvalues and Eigenvectors, respectively.

4. Sorting the Eigenvectors in descending order, with respect to the Eigenvalues, leads to the ordered principal components. Each component will describe the direction of maximum variance in the data, subject to being orthogonal to the other components. In our case, where the data are 3D points, the two first principal components will span the fitting plane. The third component will point in the direction of the plane-normal, as the variation in this direction will be minimal, compared to the two previous components. In this work, the direction of the normal is corrected such that it always points towards the sensor.

Using a point on the plane (e.g. $\mathbf{p} = \mu_{\mathbf{Q}}$) and the corrected normal $\mathbf{n}$, any point $\mathbf{q} \in \mathbb{R}^3$ can be classified as either foreground(1) or background(0). This is done using the following equation:

$$\text{IsForeground} = \begin{cases} 1 & (\mathbf{q}-\mathbf{p})^T\mathbf{n} > \tau \\ 0 & (\mathbf{q}-\mathbf{p})^T\mathbf{n} \leq \tau \end{cases}, \tag{5.8}$$

The equation calculates the signed distance from the point to the plane and compares the distance with a threshold parameter τ. Any point above the plane will have a positive distance while points below the plane will have a negative distance. The parameter τ can be used to include/exclude points close to the plane. If the value is positive, points very close to the positive side of the plane will still be classified as background. This is illustrated in Figure 5.9. It can be observed that the infant's weight affects the planar surface of the mattress. The wrongly classified mattress points are removed using a positive value of τ. The remaining data can now be used for further analysis.

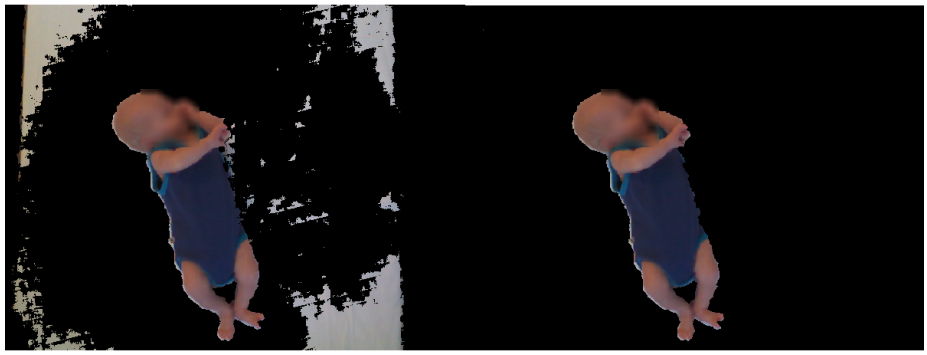


Figure 5.9: Based on fitting a plane to the mattress, points below the mattress can be removed. Left: Points exactly on or below the plane are removed. Right: Points above, but close to the plane are also removed.

5.4 Concluding Remarks

The recordings of the infants are done using a low-cost depth sensor, which captures both color and depth information. In addition, the sensor has a build in motion tracking system, but unfortunately, this cannot track infants. The color could potentially be used for tracking the skin of the infant, but this work will focus on the depth data. Later studies could utilize both modalitites to improve the results.

Chapter 6

Pose and Motion Estimation

Because the data for this study is obtained simultaneously with the project, a learning-based approach is not possible. This chapter will consider different ways of identifying the body parts of the infants, based on assumptions on the human/infant body. The learning-based approach might be revisited later, once enough data have been obtained.

6.1 Graph-based Pose Estimation

Inspired by the work in [61, 7] the first attempt on estimating the pose of the infant is based on methods from graph theory. The idea is that the extremities such as hands, feet and the head will appear as objects pointing out from the center part of the body. Based on the (segmented) depth data, every pixel can be considered as a node in a graph. Two nodes should be connected by an edge if they are neighbors and if their depth difference is small. The weight of the edge can simply be based on the distance between the respective Three-dimensional (3D) points. It should be noted that the approach is also applicable for pure 3D data. However, the neighborhood of different points can easily be extracted from the pixel locations in the depth image. Once the graph is constructed, the goal is to estimate the location of the extremities. To illustrate the concept, we consider a simple stickman figure in a Two-dimensional (2D)

image as seen in Figure 6.1. The point C is the center of the body and is assumed to be known beforehand.

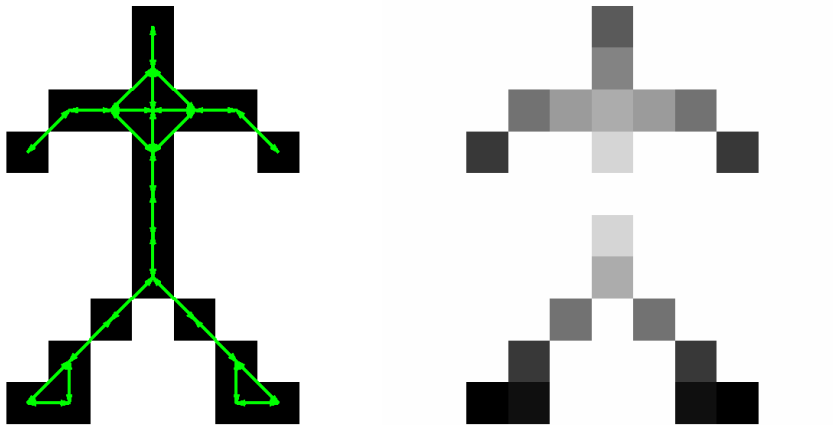


Figure 6.1: Left: An image of a stickman figure is represented as a graph and pixels/nodes close to each other are connected by edges. Right: The shortest distance from the center of the object to all other nodes is calculated and visualized (high values = dark).

1. Starting from the source node (C) in the graph, calculate the shortest path to any other node, using Dijkstra's Shortest Path Algorithm [16].
2. The next step is to sort the nodes in descending order based on the path distance. In order to locate the n extremities, a number of approaches can be applied;
   (a) Illustrated in Figure 6.2.
      i. Remove all nodes with a path distance smaller than τ.
      ii. Assuming the body parts are not overlapping/connected, do a simple k-means clustering, where k is the number of desired extremities. For each cluster, choose a candidate node as the extremity. This candidate can either be the node with the biggest path distance or the node closest to the cluster centroid.

(b) Illustrated in Figure 6.3.

i. Find the node with the longest path from C and add it to the list of extremities. For this node and all nodes being part of the longest path, the edge weights are set equal to zero. On the respective path, the half-distance node is added to another list, containing the sub-extremities. With the updated graph weights, update the shortest path distance map. This is done in order to avoid locating a new maximum right next to the previous found maximum.

ii. If the number of desired extremities is found, the algorithm stops, otherwise 2(b)i is repeated.

3. Once the extremity candidates have been found, the goal is to identify these as either left/right hand, left/right foot or head. In this study, the task is simplified, using the assumption that the infant's head-direction is known. In the following, each extremity is classified based on its direction. This direction is estimated as the direction of the path going from the root node to the extremity, not to confuse it with the beeline direction. Furthermore, the angle between two directions is used as a metric.

(a) Optional: In case the orientation of the infant is unknown, one solution to estimate this is to classify the extremities based on anatomical assumptions. The direction of the head and the hands will be similar and almost opposite to the direction of the legs. A simple clustering of the directions into two clusters might be enough to separate the upper and lower extremities. This can then be used to estimate the head-direction. The identification of the left and right side is straightforward, under the assumption that the infant is in supine position.

(b) Identify the head, as the one that points in the head-direction and remove the chosen extremity.

(c) Identify the left/right hand as the two extremities with direction closest to head direction. The two extremities are identified as the left or right hand, based on the direction pointing left or right, with respect to the known orientation of the infant. Remove the two chosen extremities.

(d) The two remaining extremities are assumed to be the feet and are identified using the same procedure as for the hands.

Based on the described approach, the skeleton of an infant can be estimated for a single depth-frame in the data. Figure 6.4 visualizes the results of four different infants. Notice how the sub-extremities are visualized as well, which indicate the estimated position of the elbows and knees.

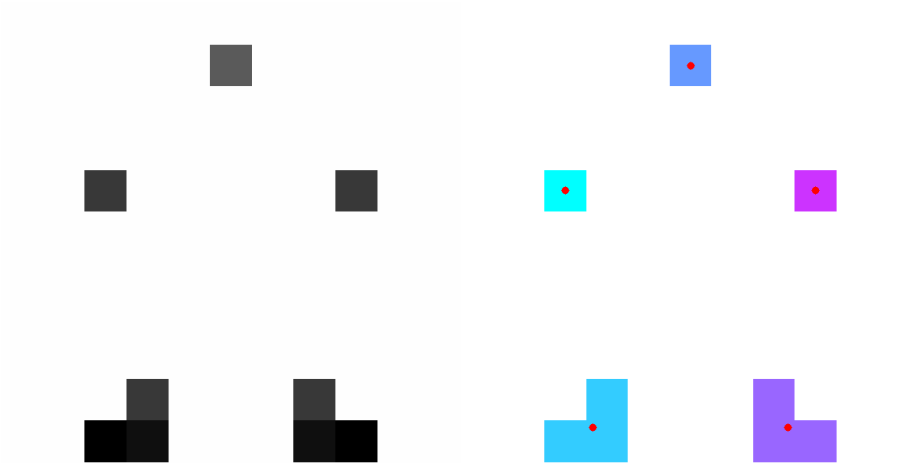


Figure 6.2: Left: Using a threshold on the path distances the body center and the surrounding pixels are ignored. Right: Using a simple k-means clustering the five extremities are located as the cluster-centroids (red dots).

As already discussed by authors of similar studies, this method fails to estimate the pose when the infant has overlapping body parts. This is illustrated in Figure 6.5, where the approach fails. One reason is that the assumptions for the graph-based method is no longer satisfied. Furthermore and more importantly, the graph-based approach does not include any constraints regarding the pose of the infant/human, except for the direction of the extremities. What happens between the root node and the extremities is ignored by the method, as long as the five extremities are found. Due to this, a more anatomical correct approach is considered in the next section.

6.2 Model-based Pose Estimation

In order to incorporate the anatomical shape of a human being and constrain the possible poses an articulated 3D model is defined as in [24, 17]. The 3D

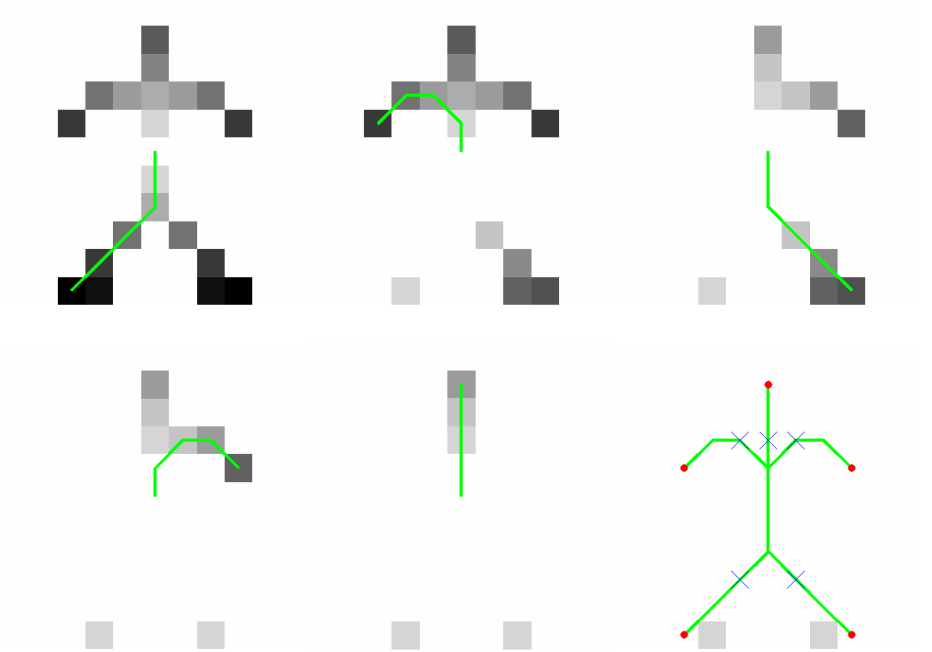


Figure 6.3: The shortest path distances are visualized for each iteration in the iterative shortest path method. In reading order the first five images includes the paths (green lines) to the five extremities. The last image (bottom-right) illustrates the location of the five extremities (red dots), as well as the half-distance nodes(blue crosses).

model represents the surface of a human person. The model is constructed from a set of 3D structures, such as ellipsoids, cylinders and spheres. The structures are connected in a skeleton structure and the root is chosen to be center of the body. In order to describe the distance between the model and the observed 3D data, the Euclidean distance is used as a metric.

6.2.1 3D Structures

In this study, the 3D model is constructed from three kinds of 3D structures, namely super-ellipsoids, spheres and cylinders. Each of these structures can be represented by various properties, describing their position, orientation and size. Furthermore, for each of the structures, a distance function is defined which

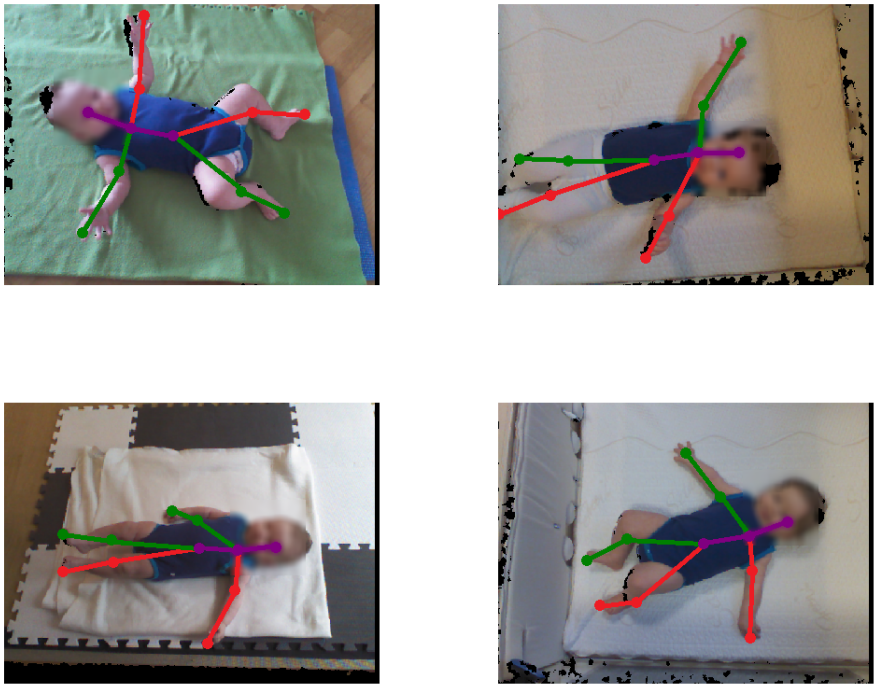


Figure 6.4: Using the graph-based approach for pose estimation, the skeleton of four infants is visualized.

calculates the distance between the structure and a 3D point. The properties and distance functions are listed below. Furthermore, each structure is listed with the associated body parts.

- Cylinder (Upper/Lower Arms + Upper/Lower Legs + Feet):
  - Length
  - Radius
  - Starting position
  - Orientation - One of the representations in Section 3.6.
  - Distance: The distance can be calculated directly, by using the formula for calculating the distance from a point to a line. Given a line defined by a starting point **a** and an endpoint **b**, the distance from a

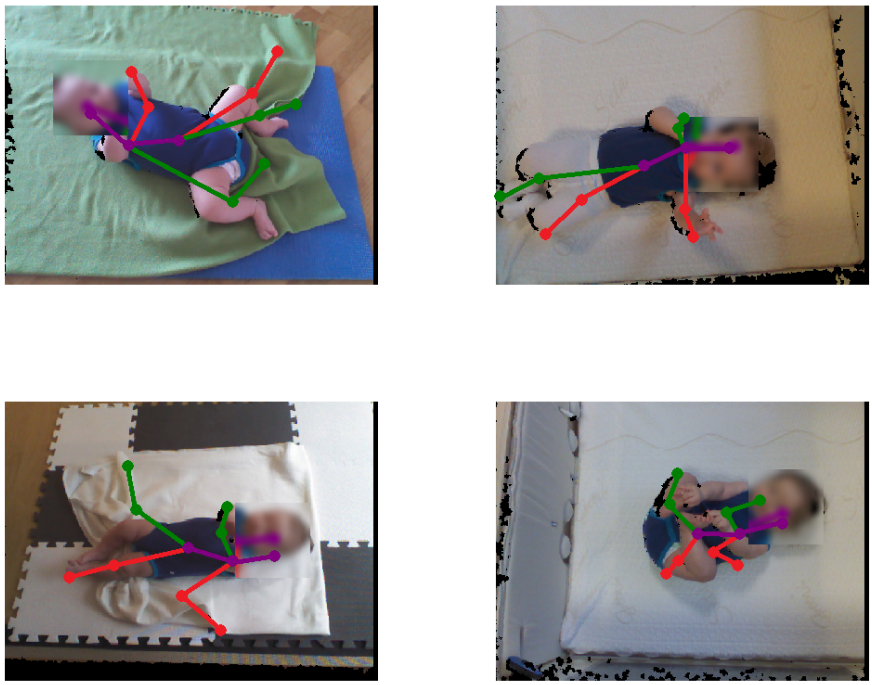


Figure 6.5: For some poses the graph-based approach fails, as the assumptions for the graph-based approach are violated.

point $\mathbf{p}$ to the line can be calculated using equation 6.1.

$$d_{line}(\mathbf{p}) = \begin{cases} \|\mathbf{p} - \mathbf{a}\| & \lambda \leq 0 \\ \|\mathbf{p} - \mathbf{b}\| & \lambda \geq 1 \\ \|\mathbf{p} - (\mathbf{a} + \lambda\,(\mathbf{b} - \mathbf{a}))\,\| & \lambda > 0 \wedge \lambda < 1 \end{cases} \tag{6.1}$$

Here,

$$\lambda = \frac{(\mathbf{p} - \mathbf{a}) \cdot (\mathbf{b} - \mathbf{a})}{\|\mathbf{b} - \mathbf{a}\|^2}, \tag{6.2}$$

is the normalized length of $\mathbf{p}$ projected onto the vector $\mathbf{b} - \mathbf{a}$. The unsigned distance from the surface of the cylinder with radius r to the point $\mathbf{p}$, is simply found as;

$$d(\mathbf{p}) = |d_{\text{line}}(\mathbf{p}) - r| \tag{6.3}$$

- Sphere (Head):

  - Radius
  - Center
  - Distance: The distance for the sphere is simply calculated as the distance between the center $\mathbf{c}$ of the sphere and the point $\mathbf{p}$, while taking the radius of the sphere, r, into account.

$$d(\mathbf{p}) = |\sqrt{(\mathbf{c}_x - \mathbf{p}_x)^2 + (\mathbf{c}_y - \mathbf{p}_y)^2 + (\mathbf{c}_z - \mathbf{p}_z)^2} - r| \tag{6.4}$$

- Super-ellipsoid (Stomach):
  - Semi-diameters
  - Center
  - Orientation - One of the representations in Section 3.6.
  - Distance: There is no closed form solution to calculate the distance from a point to the surface of a super-ellipsoid. Due to this, the problem of finding the distance is usually done using numerical/iterative methods [43, 63]. In this work, the distance is approximated using a distance map. The approach is described below.
    1. Normalize the super-ellipsoid with respect to position and orientation.
    2. Create a voxel grid around the super-ellipsoid and calculate the distance from every voxel to the ellipsoid, e.g. using an iterative approach. The resolution of the voxel grid can be increased for better approximations to the correct distance, but at the cost of increased memory usage. In this work, a resolution of $40 \times 80 \times 40$ voxels is used, which results in acceptable distance approximations.
    3. Given a new point $\mathbf{p}$, an approximation to the distance can be found by mapping the point to a voxel in the distance-map. If the point is outside the voxel grid, the distance can be set to a predefined distance or a distance based on the semi-diameters.

    The generation of the distance map is very in-efficient, but it will later be described why the distance map is appropriate in this study.

Combining the different structures, results in the full human model as seen in Figure 6.6.

6.2.2 Modelling Joints

The structures in the model are connected at the joints. Each body part is assigned a joint and the joint is always located in the "start" of the body part,

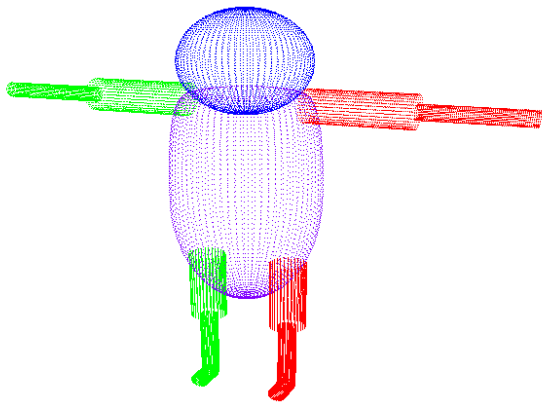


Figure 6.6: The 3D model used in this work is visualized as a articulation of different structures. The colors are only used for differentiating between left(red), right(green) and centered(purple/blue) body parts, with respect to the front direction of the model.

ordered from the center of the body. For every joint in the model a local coordinate system should be defined. The basis is defined, such that the second axis is parallel with the medial axis of the connecting body part. The bases are generated from the rotation representations described in Section 3.6. The axis angle representation is used to describe rotations, using three consecutive single axis rotations. The order of the rotations is listed below and related to the anatomical understanding of movements in joints.

1. Extension/Flexion: This is e.g. used for lifting the thighs or moving the arms forward or backward in the horizontal plane.
2. Abduction/Adduction: Describes rotations that result in sideways movements, as when the legs are split.
3. Internal Rotation: Rotation of a limb around itself.

The basis of the root joint (stomach) is chosen such that the second axis points towards the head of the model. Figure 6.7 visualizes the model with the local coordinate systems. Notice that the coordinate systems of the left and right sides are mirrored. This is done such that angle parameters belonging to opposite body parts can be compared directly, without the need to invert the angles. Furthermore, the visualized model is the default configuration of the model.

Notice how the bases of the upper arms, upper legs as well as the feet are different from the bases of the parent body parts. This is simply a choice made by the authors, such that the default configuration is represented by an all zero angle representation.

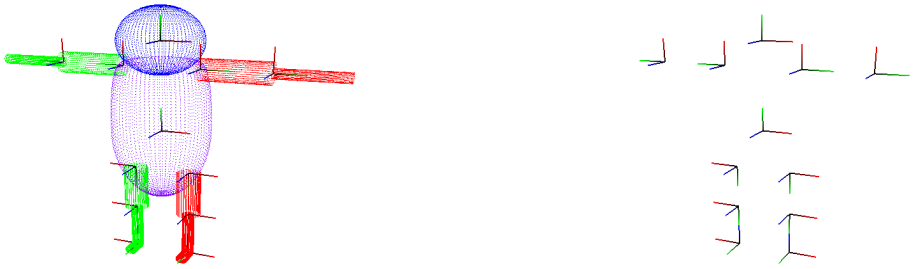


Figure 6.7: For every joint in the model, a local basis is defined. Left: The bases are visualized on the model as local coordinate systems. Right: The coordinate systems are visualized without the model. Red is the first axis, green the second axis and blue the third axis.

In addition to the bases for each joint, the joints that are connected to the stomach-structure, includes a set of parameters describing the locations of attachment. The parameters are relative to the stomach-structure. For each attached body part, three numbers indicate the relative position with respect to the stomach basis and its center. Figure 6.8 illustrates one (extreme) configuration of the attachment parameters.

6.2.3 Pose Estimation

Given the structures and their relative connections, the full model and its pose can be defined by one single vector combining all the parameters from the structures. The structures and the number of respective parameters are defined in Table 6.1. It is assumed that the size of the left and right body parts are identical.

Combining all parameters, the full model is defined by 62 parameters. However, some of these parameters can be fixed or ignored. Under the assumption that the size of the infant does not change during the recording, all size parameters and relative attachment parameters can be fixed. Fixing the size of the stomach structure is also the reason why a distance map can be used for calculating the distance from a point to the stomach-structure. As the distance map only changes due to size-changes this should only be calculated once (or every time the stomach changes size). Furthermore, a number of orientation parameters can

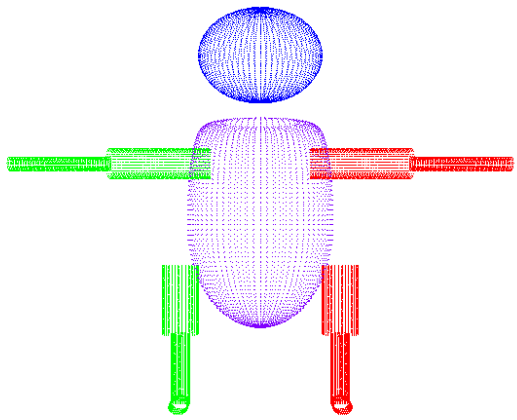


Figure 6.8: For each body part attached to the stomach-structure, a set of parameters describe the location of attachment.

Table 6.1: Overview of the parameters used for the respective structures/body parts. G means global, while R means that the parameter is related to the structure's parent structure. The numbers in the parentheses indicate the number of parameters used to represent the respective property.

Body part	Type	Position	Orientation	Size	Parameters
Stomach	Super-ellipsoid	G(3)	G(3)	(3)	9
Head	Sphere	R(3)	R(3)	R(1)	7
Upper Arm	Cylinder	R(3)	R(2 × 3)	G(2)	11
Lower Arm	Cylinder		R(2 × 3)	G(2)	8
Upper Leg	Cylinder	R(3)	R(2 × 3)	G(2)	11
Lower Leg	Cylinder		R(2 × 3)	G(2)	8
Foot	Cylinder		R(2 × 3)	G(2)	8
					Total = 62

be fixed as well, due to anatomical properties of the joints or due to symmetrical properties of the chosen structures:

- Head: The parameter describing the rotation around the second axis. As the head-structure is spherical, a rotation around the medial axis of the head does not change the shape of the model. Future work might be able to determine this rotation, in which case the parameter should be included.

- Lower Arm: As the lower arm is a cylindrical end-structure in the model, the parameter describing the rotation around the medial axis is ignored. Furthermore, due to the anatomical properties of the elbow joint, the elbow only needs one axis of rotation, describing extension/flexion of the arm.
- Lower Leg: Similar to the lower arm, the knee and elbow joints are similar in terms of possible axes of rotations. However, as the lower leg is not an end-structure, the rotation around the medial axis cannot be ignored.
- Foot: The rotation of the foot is limited to rotations around the two axes orthogonal to the medial axis.

Assuming that the size is fixed, the number of parameters that describe the pose of the model is reduced from 62 to 30. The pose of the model is thus described by a parameter vector $\theta \in \mathbb{R}^{30}$. The goal is to find an optimal parameter vector θ^* such that the sum of squared distances between the model and the observed 3D data $\mathbf{Q}$ is minimized:

$$\theta^* = \arg\min_{\theta} \sum_i d(\mathbf{Q}_i, \theta)^2 \tag{6.5}$$

Here, $d(\mathbf{Q}_i, \theta)$ is a function that returns the Euclidean distance between the model and the point $\mathbf{Q}_i$. This distance is equal to the smallest distance to the structures in the model. In order to solve the problem in Equation 6.5 and find the set of parameters that fits the 3D model to the data, we choose to do this using an iterative optimizer. This will be explained in the following section.

6.2.3.1 Levenberg Marquardt

The Levenberg-Marquardt (LM) algorithm [51] is a very popular choice in computer vision for solving non-linear least squares problems and this is exactly the kind of problem that should be solved. A general description of the LM algorithm is described, but it easy to see the similarity between Equation 6.5 and Equation 6.6. The problem is to find a parameter vector θ^* such that the sum of squared residuals between the output of a non-linear function $\mathbf{f}(\theta)$ and the desired output is minimized;

$$\theta^* = \arg\min_{\theta} \sum_i (x_i - f_i(\theta))^2 \tag{6.6}$$

$$= \arg\min_{\theta} ||\mathbf{x} - \mathbf{f}(\theta)||^2 \tag{6.7}$$

$$= \arg\min_{\theta} ||\mathbf{r}(\theta)||^2 \tag{6.8}$$

Given an initial parameter vector, the LM algorithm is an iterative method that updates the parameter vector θ while minimizing the objective function $||\mathbf{r}(\theta)||^2$. The updated objective function is thus:

$$||\mathbf{r}(\theta + \delta_\theta)||^2 = ||\mathbf{x} - \mathbf{f}(\theta + \delta_\theta)||^2 \tag{6.9}$$

The update step can be approximated by a first order Taylor approximation.

$$\mathbf{f}(\theta + \delta_\theta) \approx \mathbf{f}(\theta) + \mathbf{J}\delta_\theta, \tag{6.10}$$

where $\mathbf{J}$ is the Jacobian matrix of $\mathbf{f}(\theta)$, i.e.

$$\mathbf{J}_{i,j} = \frac{\mathbf{f}_i(\theta)}{\partial \theta_j} \tag{6.11}$$

Substituting $\mathbf{f}(\theta)$ in Equation 6.9 with the approximation in Equation 6.10, the problem of finding the update vector δ_θ can be found as the solution to a simple linear least squares problem:

$$\delta_\theta = (\mathbf{J}^T\mathbf{J})^{-1}\mathbf{J}^T\mathbf{r} \tag{6.12}$$

The derivation of this is shown in Appendix A.1. A damping parameter λ is introduced resulting in the final LM update step:

$$\delta_\theta = (\mathbf{J}^T\mathbf{J} + \lambda\mathbf{I})^{-1}\mathbf{J}^T\mathbf{r}, \tag{6.13}$$

Here, $\mathbf{I}$ denotes the identity matrix, i.e. λ is added to the diagonal elements of $\mathbf{J}^T\mathbf{J}$.

For small values of λ, the method will be close to the Gauss-Newton method and when λ is big, the solution converges to the solution of the gradient-descent method. The value of λ is usually adapted during the iteration process of the LM algorithm and different approaches for updating this value are described in the literature.

6.2.4 Initialization

One problem with the LM algorithm is that the optimizer might end up in a local minimum. In order to get a good solution, the initial guess of the parameter-vector must be fairly close to the global minimum. For this study, the pose estimation is initialized using the graph-based method described in Section 6.1. This method gives a good initial guess on the spatial positions of the extremities and combined with inverse kinematics (see Section 3.5), a

corresponding parameter-vector can be found. Using this approach, an initial guess is estimated for the colorless data in Figure 6.9. Even though the legs are not correctly aligned, the model is relatively close to the correct pose. As mentioned previously the method has some problems for poses with overlapping limbs, but this problem will be revisited in Section 6.3.3.

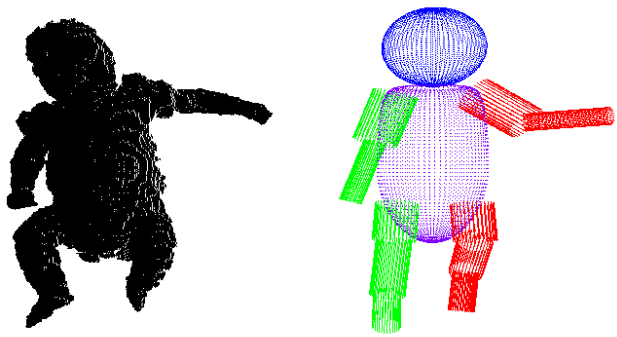


Figure 6.9: Model obtained using inverse kinematics and candidates for the position of the extremities. The extremities are found using the graph based approach.

Under the assumption that the initialization is correct, the estimated pose from one frame can be propagated to neighboring frames, if the sampling-rate is relatively small. In this work, where the data comes from the Kinect sensor, the sampling rate is 30 frames per second. The poses from neighboring frames are thus used as initial guesses for new frames. However, the graph-based approach can be used for re-initialization, if the distance between the 3D model and the data exceeds some threshold.

6.2.5 GPU Acceleration

As the optimization is done using an iterative approach, this is usually very time consuming, depending on the size of the data. The most time-consuming part is the calculation of the residual vector and the Jacobian matrix. In some cases, the Jacobian matrix is not calculated analytically, but approximated using finite differences. This is e.g. the case in this study. As the approximation is based on multiple independent calculations of distances, this can be done in parallel.

Due to this, the calculations are done on the Graphics Processing Unit(GPU) and the different steps are explained below.

- Residuals: For every 3D observation/thread on the GPU, calculate the distance to every structure of the 3D model. The minimum distance will correspond to the distance from the observation to the model.
- Jacobian: The estimation of the Jacobian is done using numerical differentiation, i.e. each column in the Jacobian matrix is approximated by a change in the residual vector, given a small change in the parameter vector. For the straightforward method, this would require that the full residual vector is calculated for every parameter. However, the Jacobian is often sparse, as the parameters does not affect all structures in the 3D model. The parameters of the foot will e.g. only affect the residuals of the observation points in the foot region and should not affect points close to the head. This does not always hold if an observation is assigned another structure during the Jacobean estimation. However, the estimation-error is minimal and the speedup gained from only calculating the non-zero elements in the Jacobian is significant better. Furthermore, the errors from the false assignments will disappear in the next update step of the LM algorithm.
- Host↔GPU Transfer: The size of the Jacobian matrix in Equation 6.13 is $m \times n$, where m is the number of observations, i.e. the number of 3D points, while n is the number of parameters. In most cases $m \gg n$, as one objective of motion tracking is to describe the data in a lower dimensional space. In this work, $m \approx 30.000$ and $n \approx 30$. In the naive setup, the full Jacobian matrix is transferred to the host device, in order to calculate the update vector. However, the amount of data transferred can be greatly reduced, by doing the calculations with the Jacobean on the GPU instead of the host CPU. From Equation 6.13, it can be observed that the Jacobian matrix is used to calculate the $n \times n$ matrix $\mathbf{J}^T\mathbf{J}$ as well as a $n \times 1$ matrix $\mathbf{J}^T\mathbf{r}$. By calculating these matrices on the GPU, the amount of data that needs to be transferred is reduced significantly.

6.2.6 Single Frame Pose Estimation

Based on the model-based approach the pose can be estimated by fitting the model to the observed data. This is done for a single frame and the estimated pose model can be seen in Figure 6.10. The initial starting guess is found using the graph-based approach together with inverse kinematics. It can be observed that the fitted model correctly estimates the pose of the infant.

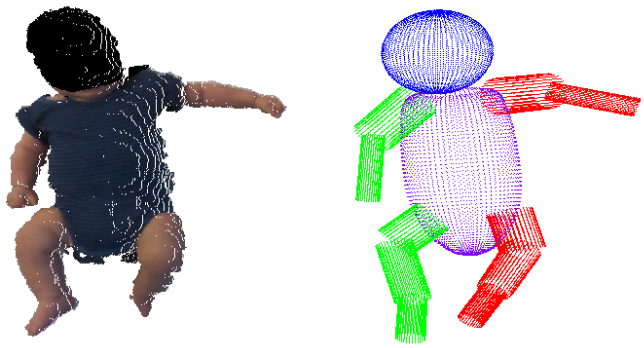


Figure 6.10: The resulting pose estimated on a single frame in the dataset.

One thing that has been not been considered in detail is the size parameters. These have been assumed fixed during the optimization step, but the question is how they are initialized. Even though infants are small, they do grow a lot during the first months after birth, meaning that a one-size model does not exist. However, there are still certain limitations and it is possible to define a model that is close to the mean size of infants. The goal is thus to find deviations from the mean-model to fit each infant and this can be done using different approaches:

- The size parameters can be measured directly from the infant. This requires some additional work doing the recording, but the size parameters will be very close to the true size of the infant.
- Before the pose estimation/motion tracking begins, the size parameters can be adjusted manually, in order to make the model and the data similar, with respect to size.
- Unfixing the size parameters during the optimization process makes it possible to automatically find appropriate size parameters. However, this requires that the model pose is close to the correct pose, as the increased Degrees of Freedom (DoFs) also increases the number of local minima.

It should be noted that a combination of the above-mentioned approaches can be used, e.g. by manually adjusting the size parameters after the optimization. This solution has been used in this study.

6.3 Pose Statistics

During this project, a relative large set of infants (72) has been recorded. The ages range from 2 to 26 weeks Corrected Age (CA) and for some infants, multiple recordings have been carried out, with one month time difference between the recordings. Figure 6.11 gives an overview of the age of the infants as well as the number of infants recorded multiple times. It should be noted that an infant recorded three times, is not included as an infant recorded one and two times.

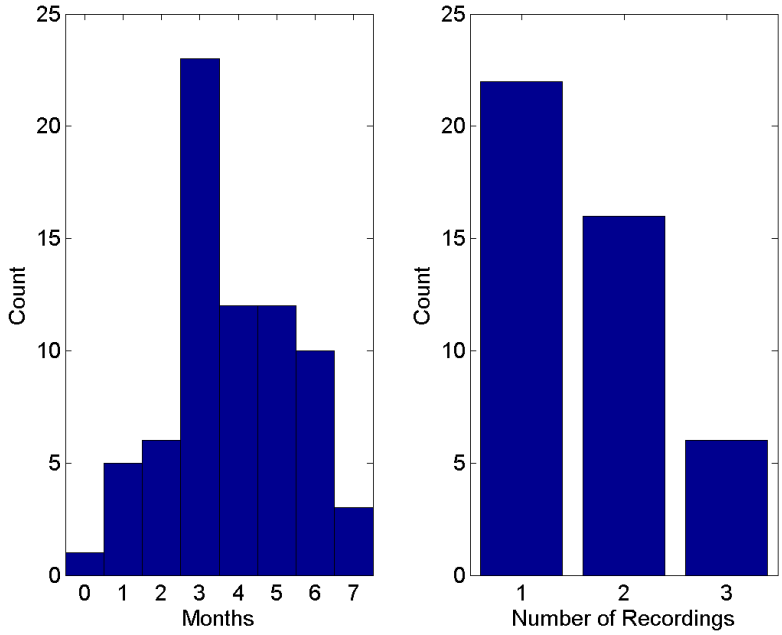


Figure 6.11: Left: The distribution of age for the recordings. Right: Some infants have been recorded multiple times and the distribution is illustrated here.

6.3.1 Joint Statistics

Considering the raw joint angles in the data, the axes-angles can be visualized, resulting in an overall view of the individual angles. This is done for the upper arm angles in Figure 6.12 (the data from the left and right arm are combined). This gives an overview of the angles used to describe the different poses of the

upper arm. For motion tracking purposes, the joint angles of a new pose can be compared to the histogram distributions. The likelihood of the pose can thus be estimated.

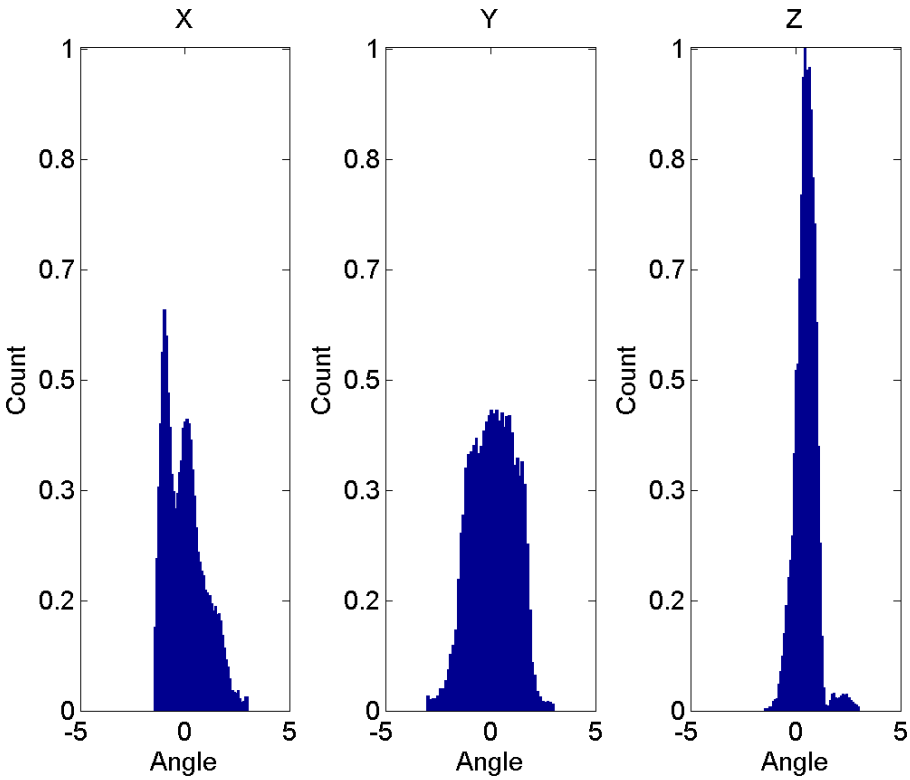


Figure 6.12: Histogram of the upper arm angles, for each of the three rotation axes.

Even though this gives a good idea of the validity of the pose, this assumes that the joint angles are uncorrelated, which might not necessarily be the case for all joints. One example is the correlation between the hip joint and the knee-joint. If the thigh is lifted from the ground, this usually results in a flexion of the leg. This correlation is visualized in Figure 6.13. It is seen that low thigh-ground angles, meaning that the thigh is lying on the ground, also results in a smaller angle between the thigh and the leg. When the infants lift their thighs, the legs are flexed and the angle between the thigh and leg increases. Furthermore, it can be seen that the thigh-ground angle does not exceed an angle of 140 degrees. This is reasonable, as larger angles would mean that the thigh points towards the head of the infant. This is of course possible, but it is more likely that the

thigh is pointing away from the head, i.e. the angle is below 90 degrees and this behavior can also be seen in the figure.

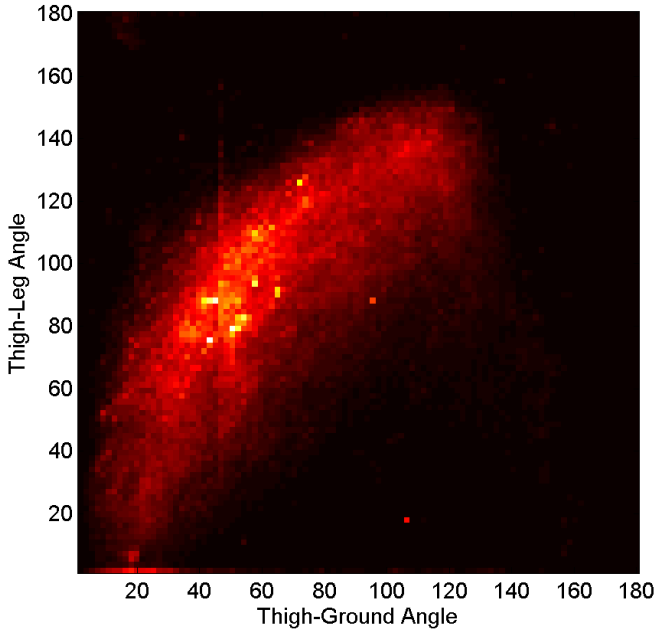


Figure 6.13: The correlation between the thigh-ground angle and the thigh-leg angle is visualized (black is low and yellow is high). Observe that when the thigh is lifted, this results in a flexion of the leg.

6.3.2 Pose Clusters and Variation

Working on all pose parameters in combination, multivariate analysis can be used for extracting common poses from the data. In order to find the most frequent poses throughout the whole dataset, a K-means clustering is applied to the data, where $K = 20$ clusters. Methods for choosing K exist, but a cluster size of 20 was chosen, as the interesting part for this analysis was to locate dominating clusters. However, as five months old infants can do poses not capable for a two weeks old infant, the data is divided into three age groups. Figure 6.14 visualizes the three age groups and their four biggest cluster-centers. Observe that older infants tend to have more poses that represent joining the hands and feet. This is expected, as older infants are often stronger and have

more control of their body and they also tend to play with their feet in a more coordinated pattern.

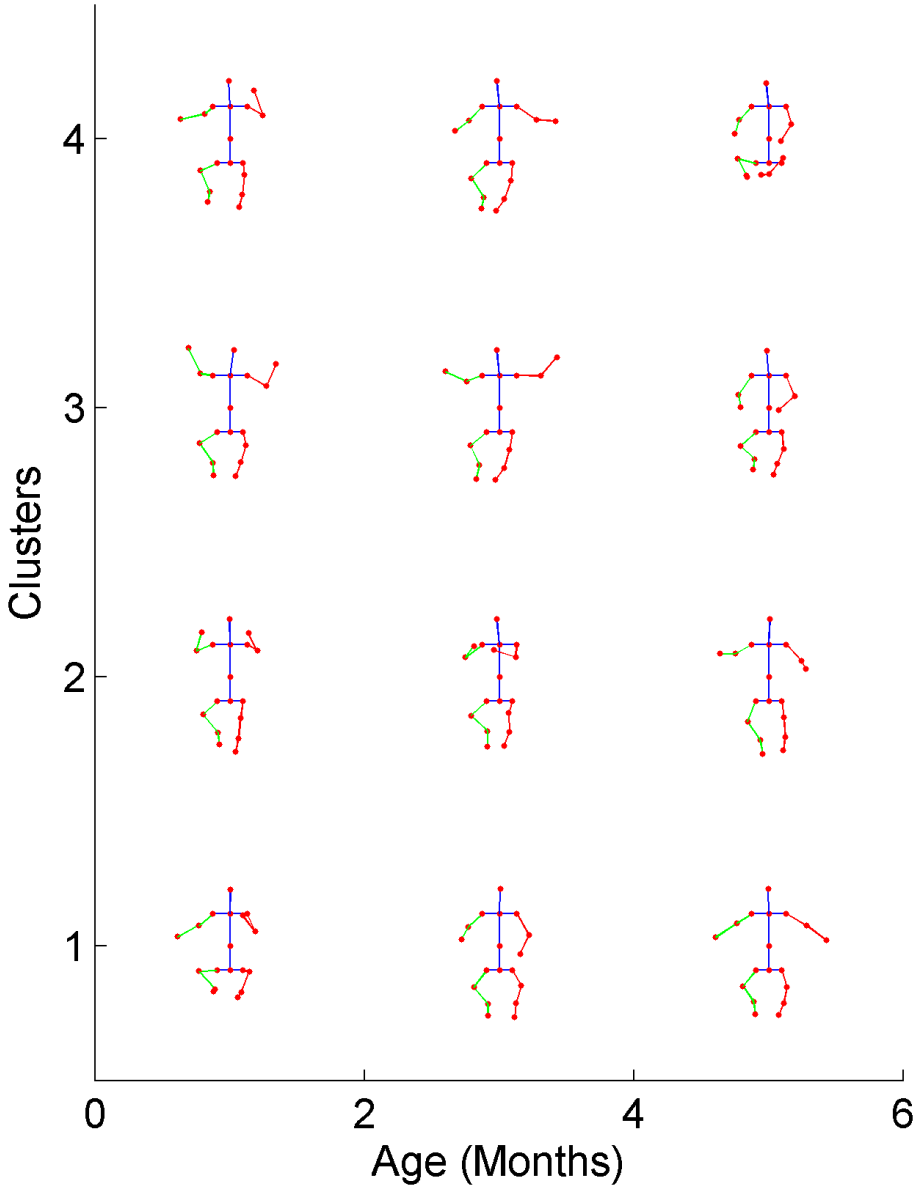


Figure 6.14: Clusters of the pose data for different ages. Each column contains the 4 biggest clusters for the respective age group.

In order to describe the variation in the poses for different ages, Principal Component Analysis (PCA) is used to model the variation along the principal components. Figure 6.15 illustrates the variation of poses along the first principal component, for different age groups. Here, the difference between the different age groups is much lower, compared to the cluster analysis. It can be observed that for all three age groups, the first principal component primarily describes if the hands and legs are joined or spread out. However, the older infants tend to have more of the extreme cases, where hands and feet are joined together. The variation described by the first principal component in the three age groups are respectively 19, 23 and 37% ordered from young to old. This might be an indication that the younger infants have less control over their movements, with respect to coordination between the different body parts, while the older infants more often do goal specific poses/movements.

6.3.3 Statistical Pose Estimation

Based on the results from the cluster analysis, the biggest clusters describe the most common poses of infants. These results can be used for initializing the pose for a new unknown frame. As described in Section 6.1, the graph-based method is capable of estimating the pose, under the assumption that the extremities clearly extends out from the stomach body part. If this is not the case, the estimation of the pose is no longer valid. However, using the pose database, the most common poses are good candidates for the true pose of the infant, even when the infant's body parts are overlapping. Figure 6.16 illustrates a few examples for the best common pose fit, which results in acceptable pose estimation. It can be observed that the orientation of the different body parts are not perfect, as all variation is not included in the common pose data. The results are based on a uniform weighting of the common poses, i.e. all poses are equally likely to be the correct pose. However, the approach could be extended with age-specific weighting, such that poses common to the age is more likely to be the correct pose. Another extension could include adapting the weightings over time, such that some poses become more likely for the specific infant after a few poses have been estimated.

6.4 Concluding Remarks

Two methods for estimating the pose of an infant using depth data have been considered. The first method is based on representing the data as a graph and classifying body parts based on geodesic distances from the body center. The

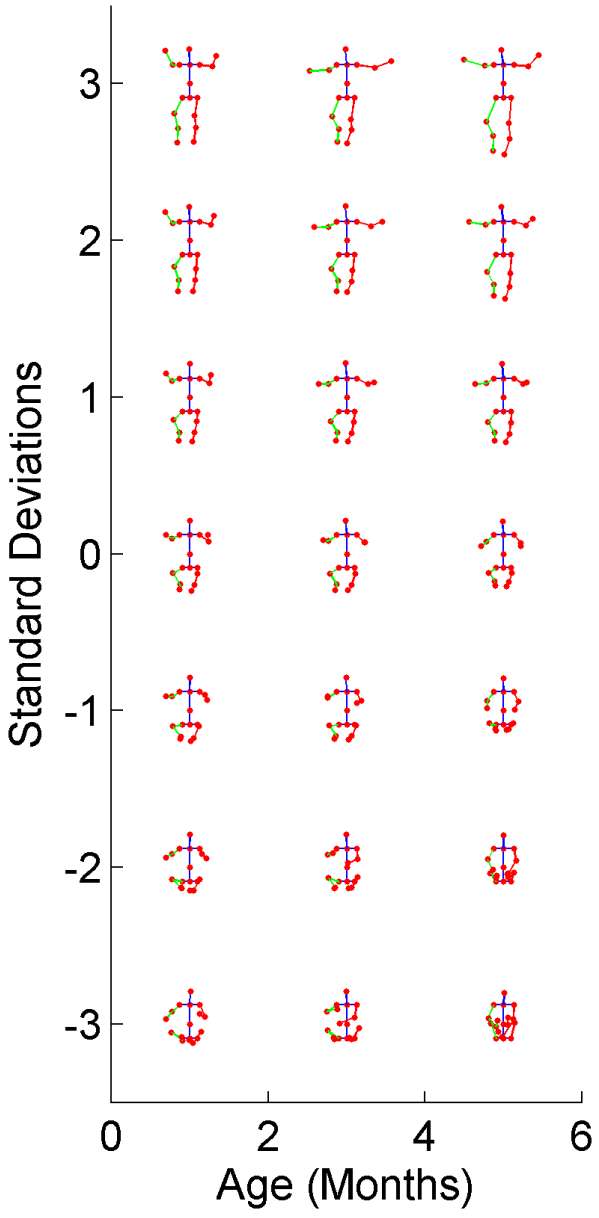


Figure 6.15: Visualization of the variation along the first principal component for different ages.

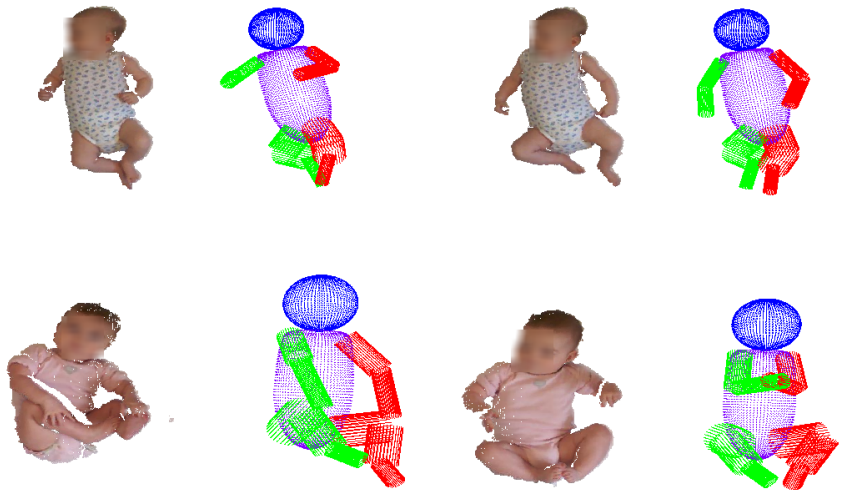


Figure 6.16: Based on the most common poses, the pose for two new infants is initialized as the pose that minimizes the residuals between the 3D model and the data.

method is fast and works well when the infant's limbs are stretched out. However, in cases where multiple body parts overlap or intersects, the method fails due to the underlying assumptions. The second approach solves this problem using a more constrained description of the body. Here, the approach models the full skeleton and surface of an infant, which gives an anatomically closer representation of the body. The model-based approach is used on all the recorded data and an infant pose database has been generated. From this, machine learning is used to extract common pose variation. In addition, a simple example for using the data for learning-based pose estimation is given.

Chapter 7

Feature Extraction

The problem of finding and modeling the motion of the infant was solved in the previous chapter. However, the results are represented as single pose estimates, represented as a combination of joint angles. The results should thus be converted to features similar to those of classical motion tracking systems.

7.1 Pose Features

For each frame f in the data, the pose estimation is described by a parameter vector θ_f. The elements in the parameter vector θ_f consists of a global position and orientation as well as local joint angles. For a full video, the parameter vectors can be combined into a full matrix, where each column relates to a single frame:

$$\mathbf{\Theta} = [\theta_1 \dots \theta_f \dots \theta_n] \tag{7.1}$$

For every frame, the spatial positions of the body parts can be calculated. Furthermore, based on the associated rotation matrices, for every joint, a direction-vector can be calculated as well. The direction can be either global or local.

The global orientation considers all parent joints and obviously, every joint is affected by the global orientation. The local directions are based on the directions based only on the local angles of a single joint. From the single-frame features, the distances and angles between different body parts are calculated. This can e.g. be used to indicate if the infant reaches for its feet, mouth, opposite hand etc. as illustrated in Figure 7.1.

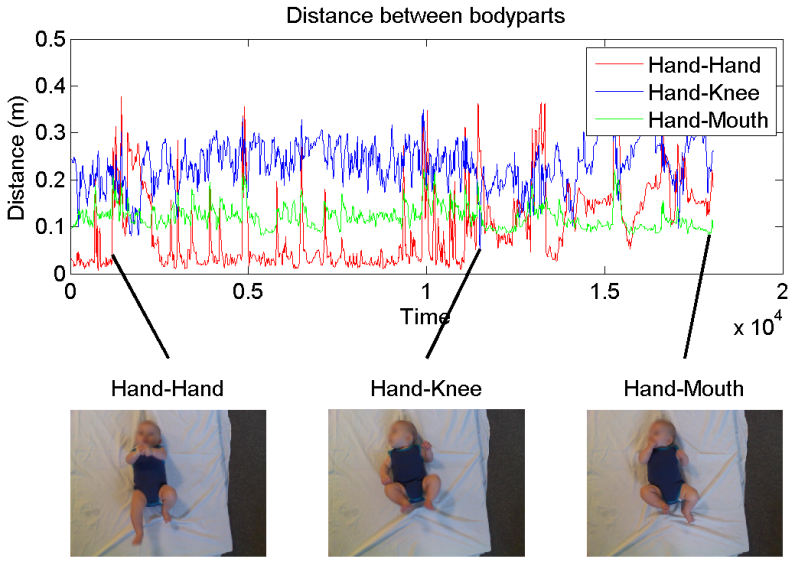


Figure 7.1: The distance between different body parts can be used for getting a better overview of the infant's poses.

7.2 Motion Features

In order to describe the motion of the infants, it is not enough to look at a single frame, but instead multiple frames should be included, in order to estimate quantities such as velocities and accelerations. As for the pose features, this can both be computed from the spatial features and from the angle parameters. The quantities can simply be estimated as the first- and second-order differences between successive frames:

$$\mathbf{v}_f = \mathbf{p}_f - \mathbf{p}_{f-1} \tag{7.2}$$
$$\mathbf{a}_f = \mathbf{v}_f - \mathbf{v}_{f-1}, \tag{7.3}$$

where $\mathbf{p}_f$ is the spatial or angular parameters for frame f.

In the case of non-uniform sampling of the data, it might be appropriate to scale the quantities with respect to the time-difference between the frames. This makes the quantities comparable with each other and makes the features more general, in the sense that they can be compared with other systems. In Figure 7.2 the velocities of the left and right hands of two infants are illustrated. It can be seen that the first infant has a more symmetrical movement pattern, in the sense that velocities for the two extremities are similar in both magnitude and frequency. However, the second infant shows a dominant movement pattern for the left hand as well as more movements with higher velocity. This kind of plot might indicate that the infant should be stimulated more in the right side. However, this conclusion should not be final, but the data should be evaluated by a person familiar with normal infant movement.

7.3 Scene Flow Estimation

Until now, the analysis has been done on the features extracted from the motion tracking. This approach is able to model the gross motion of the infants, but at the cost of ignoring finer motions. Currently the foot and lower arm is modeled as one single cylinder. Small movements in the toes/fingers will not necessarily be modeled by the motion features, even though the motion is apparent in the recorded data. Due to this, the depth data is revisited in order to extract more detailed motion features from the data. In image analysis, a concept known as optical flow is used to densely track pixels between Two-dimensional (2D) frames. With the recent introduction of various RGB-D sensors, optical flow has been adapted to this kind of data and the concept is named scene flow [85]. For a frame f the goal is to find the displacement field $\mathbf{D} \in \mathbb{R}^{n\times 3}$, such that the Three-dimensional (3D) data $\mathbf{X}^{f-1}$ from the previous frame is aligned with the current 3D data $\mathbf{X}^f$. Mathematically, this can be written as:

$$\mathbf{D}^* = \arg\min_D \sum_i ||\mathbf{X}_i^f - (\mathbf{X}_i^{f-1} + \mathbf{D}_i)|| \tag{7.4}$$

However, this is under the assumption that the number of data points does not change between the two frames and that there is a perfect match for every

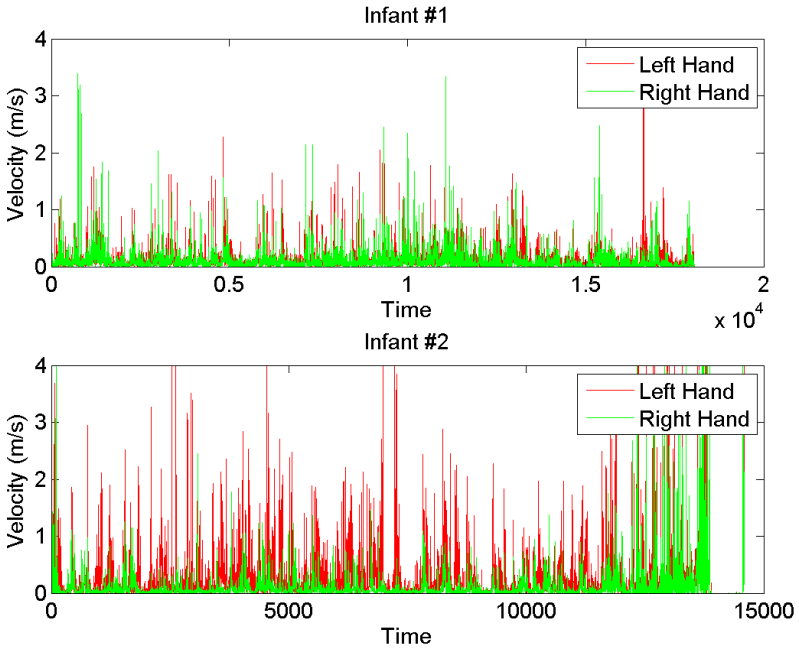


Figure 7.2: The velocities of the hands for two infants. Comparing the two infants, infant #1 have more symmetry between left and right hand, while the left hand dominates the movements for infant #2. In addition, infant #1 have less high velocity movements, compared to infant #2.

point. This is rarely the case and thus correspondences between the two point clouds should be estimated. In optical flow, this correspondence is usually based on similarity metrics in the intensities. For scene flow the correspondences is usually based on matching the local shape of points. For both optical flow and scene flow, the results are regularized, such that the neighborhood between pixels/points are preserved. The goal is thus to define a function that finds the point in $\mathbf{Y}$ that corresponds the point $\mathbf{X}_i$:

$$j = \mathrm{M}(\mathbf{X}_i, \mathbf{Y}) = \underset{j \in 1 \ldots |\mathbf{Y}|}{\arg\min}\, \mathrm{d}(\mathbf{X}_i, \mathbf{Y}_j) \tag{7.5}$$

The function $d()$ simply defines a metric between the points. The function could be the Euclidean distance function between the points, but the metric should somehow incorporate the shape information. Equation 7.4 can be adapted with the matching function in Equation 7.5 in order to be able to match points between two datasets. The matching function can be based on several features extracted from the data. One approach is considered in the following section,

where the results from the motion tracking is used to match points between frames.

7.3.1 Motion Guided Scene Flow

Using the results from the motion tracking, the 3D data can be segmented by the assigned body parts. Furthermore, it is known how the respective body parts moves from frame to frame. This can be used for densely tracking the 3D data. In Figure 7.3, the approach of aligning body parts between frames are illustrated. Based on normalizing the data with respect to the orientation and position of the body parts, points from different frames can be aligned.

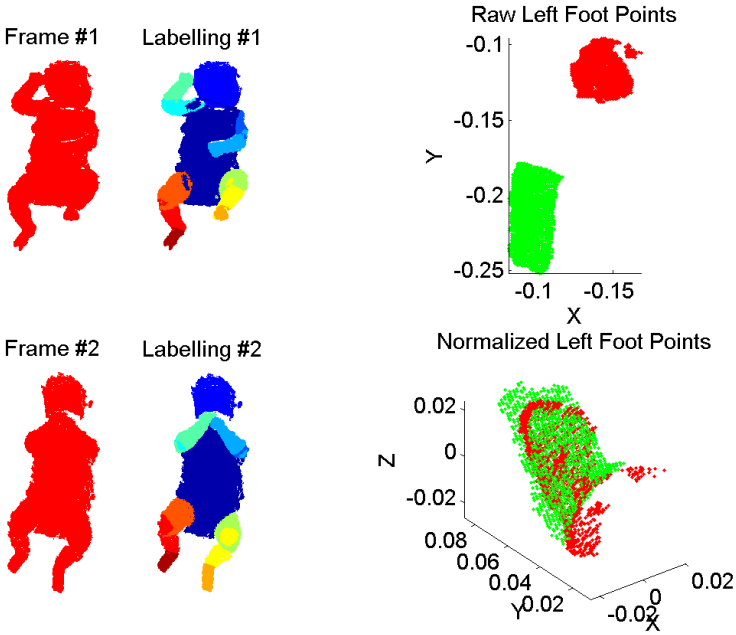


Figure 7.3: First column: The point cloud from two different frames are illustrated. Second column: Based on the 3D body model, the points are classified as the closest body part in the model. Third column top: The points assigned to the left foot are considered and there is an obvious difference in both orientation and position. Third column bottom: Normalizing the points with respect to the known orientation and position of the body part aligns the two point clouds.

The final correspondence can simply be based on a nearest neighbor search between the two datasets. The solution can be extended with the Iterative Closest Point [68] algorithm, in order to further align the two datasets. Irrespective of the further alignment, the resulting correspondences can be used to match (non-normalized) data between two frames. In Figure 7.4, the flow in the XY-plane is illustrated, based on the two frames in Figure 7.3.

Figure 7.4: Visualization of the 2D flow between two frames. The points are colored based on the direction of the flow, while the saturation represent the magnitude of the flow. The color-wheel helps the reader understand the relation between color and direction. The flow is consistent with the motion seen between the two frames in Figure 7.3.

It should be noted that the resulting flow is global in the sense that the approach estimates the flow between two frames, without considering the dependencies between limbs. A motion in the thigh results thus in a flow for the thigh, as well as the leg and foot connected to the thigh. Local scene flow is therefore intro-

duced, which is computed as the flow between the data, normalized by the body parts. Here, body parts are normalized with respect to their parent limbs. In Figure 7.5, the magnitude of the global and local scene flow is visualized. Using the local scene flow, it is easier to observe flows concerned with the individual body parts. This can e.g. be seen in the right foot or the left hand.

Figure 7.5: The scene flow is visualized as normalized magnitudes (dark = large flow, white = small flow). Left: Magnitude of global scene flow, similar to the flow visualized in Figure 7.4. Right: Visualization of the magnitude of the local scene flow. The discontinuities in the local scene flow comes from the fact that points are assigned to different body parts.

The scene flow gives a dense description of the motion of every point on the body. The previously described pose and motion features could be visualized as time series, which gave a good overview of the full sequence. However, for the scene flow, this is not as easily visualized. The scene flow could be integrated over the whole sequence. However, simply integrating the projected scene flow as seen in Figure 7.5, does not necessarily give a better understanding of the infants movements, but will result in a mix of scene flow from different body

parts. In order to better distinguish the different body parts, the scene flow can be projected onto the underlying skeleton model, as seen in Figure 7.6. This makes it possible to normalize with respect to pose and directly compare two frames or data from two recordings. Some information is of course lost in this process, as the scene flow is projected onto a single line.

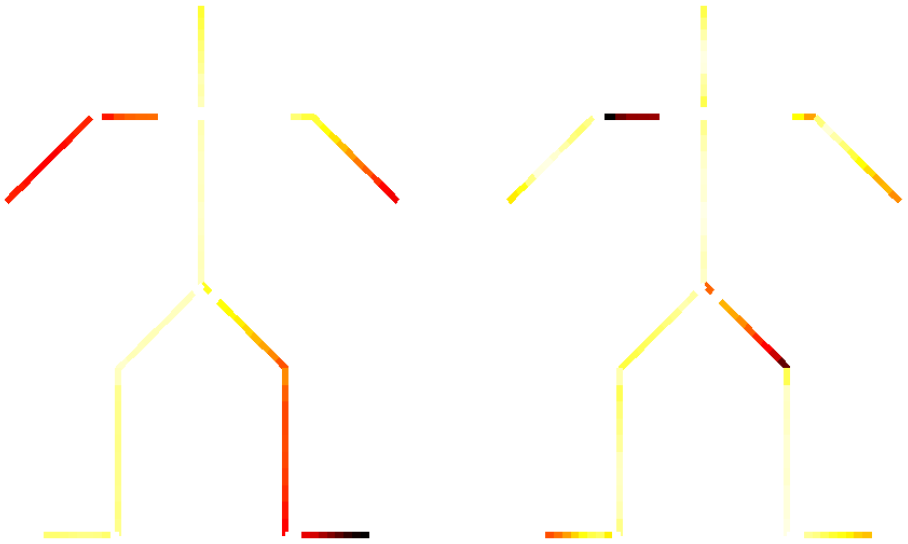


Figure 7.6: Projecting the scene flow onto a normalized skeleton structure, the scene flow on the different body parts can observed. Notice the similarity in magnitude with Figure 7.5.

Going back to the original problem of integrating the flow over a full sequence, this is done for one infant. Figure 7.7 illustrates the results for both the global and local flow. From the colored skeletons, it can be observed that the integrated amount of movement in the right arm is relatively small, compared to the other body parts. For comparison, this infant is the same as considered in the bottom figure in Figure 7.2.

7.4 Concluding Remarks

Using forward kinematics (Section 3.5), the position of the respective body parts/joints can be estimated. This results in features similar to the measurements from optical marker-based motion tracking systems. Furthermore, as the locations are uniquely defined, features such as velocity and acceleration can be

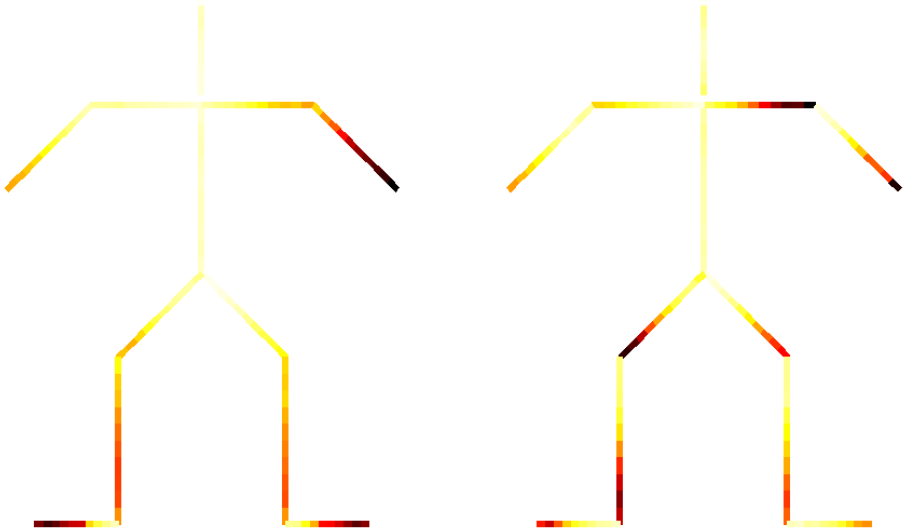


Figure 7.7: Integrating the (global) flow over the full motion sequence, results in an overall view of the infants movements in the different body parts.

estimated as well, resulting in features similar to those of non-optical motion tracking systems. Moreover, the dense depth images are revisited in order to get even denser estimations of the infant's body movements. This can be compared with the freedom of positioning markers anywhere on the body during a marker- or sensor-based motion tracking scenario. Future studies could focus on comparing the feature estimations with the true data obtained from marker- and sensor-based systems.

Chapter 8

Classification

Given the different features extracted from the data, the final step is to classify the motions and behavior of the infants. Different examples for classifications are considered, utilizing different modalities from the data.

8.1 Pose Classification

Based on the collected amount of estimated poses, different examples for using the data for classification is considered, with focus on infant motor development. This could e.g. be joining the hands, reaching for the feet, lifting the legs, etc. Other features describing symmetry between the left and right side can also be considered. The poses could be classified as one of the cluster-centers found in Section 6.3. However, instead certain features are defined that describe the specific motor milestones. As in Section 6.3, the infants are divided into three age groups. This is done to illustrate age-specific changes, as the increased strength of older infants, should influence the occurrence of certain poses. The different milestones are described as simple thresholds on the features extracted in the previous chapter. Joining particular body parts are based on the respective joints being within a certain distance of each other. Angular features are e.g. used for determining if the infant is rolling from side to side. The results can be seen in Figure 8.1, where the occurrence of the different milestones are shown.

The results are based on the mean value for each age group and the values are normalized to be the number of occurrences within one minute. It can be seen that for most of the milestones, the older infants scores the highest occurrences, as expected.

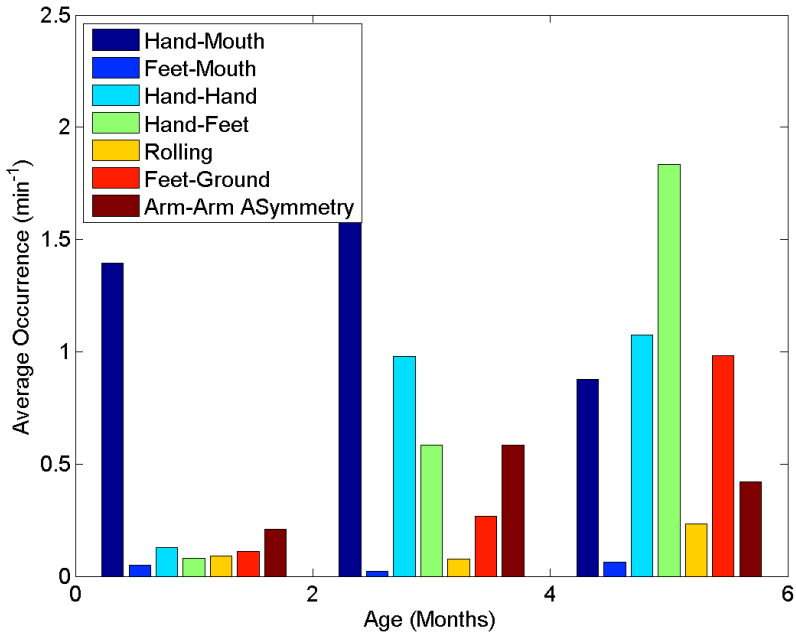


Figure 8.1: Based on distances between body parts, a number of milestones can be described and used for classifying the poses in the data. The occurrence of different milestones are summarized for three age groups.

As an example, the mean distance between the hands and the feet is plotted for all infants in Figure 8.2. A clear trend is observed in the sense that older infants are able to reach for their feet. For some of the older infants, the hand-to-feet pose is not seen, but this is acceptable. What is important is that with increasing age, the minimum distance between the hands and feet decreases.

8.2 Color-Based Classification

In Chapter 5, the color data was shortly used for finding the bodystocking that some of the infants wore. However, this was not a necessity for the later pose

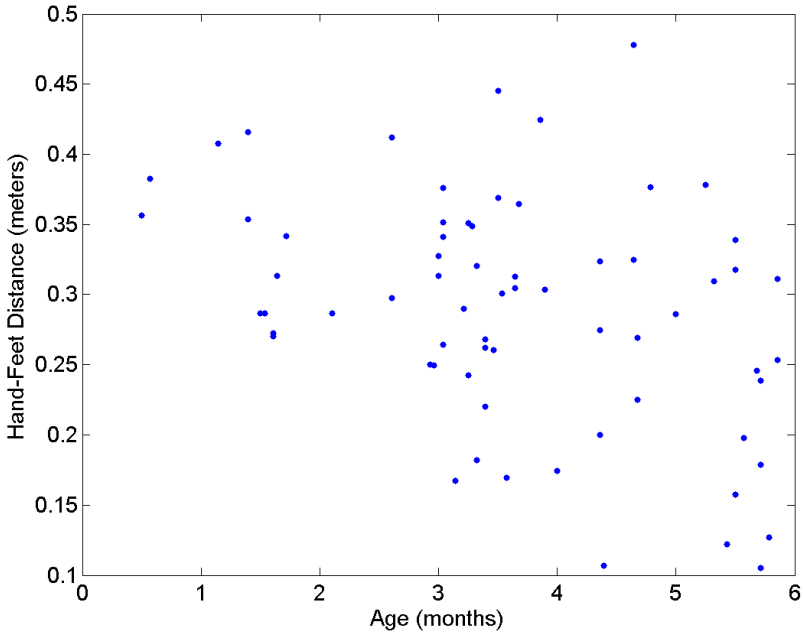


Figure 8.2: Illustrating the hand-feet distance for the different infants shows that older infants is more capable of reaching for their feet, compared to the younger infants.

estimation and the color analysis was ignored. Until now, most focus has been on the depth data, which is the primary data used for both pose estimation and feature extraction. However, it might be relevant to reconsider the color data and combine this with the results from the motion tracking. The results can be used for extracting colors from different body parts, e.g. for extracting the color of the body stocking or the skin color of the infant. This could be used for making a statistical analysis of the most popular colors of bodystockings, but the results would not be very relevant for this study. However, one thing that could be relevant is to analyze the color of the infant's face. As mentioned in Section 2.3.1, the General Movements (GMs) are affected if the infant is crying. When this happens, the infant's face turns red, relative to the usually color. This is illustrated in Figure 8.3. Each column in the image represents the mean color of the infants head for the respective frame. The first row is the raw color data, which is dominated by a Caucasian color. Normalizing the colors with respect to saturation, gives a better contrast between the extreme frames, as seen in the second row. The third row is a simple heat-map visualization based

on the amount of red in the head. This is also normalized with respect to the overall colors during the recording. The peaks, seen as the red regions, indicate frames with increased red. This might be due to the infant crying, which is indeed the case for some of the frames. For other frames, the increased redness is due to the infant doing some exhausting movement.

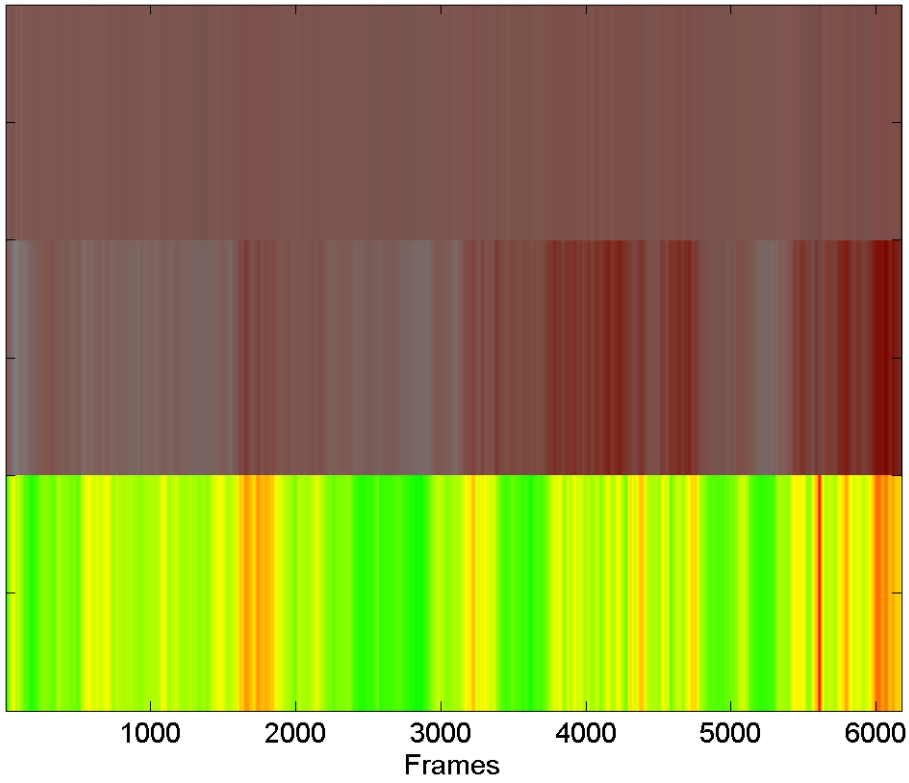


Figure 8.3: The color of an infant's face is visualized over time. Top: The raw color is visualized. Middle: Normalizing the colors with respect to the infants median-color, results in a better visualization of the color changes. Bottom: A heatmap visualizes the relative saturation of red in the face, with respect to the highest and lowest saturations.

8.3 Kick Detection

The GMs are not apparent in the infants all the time (See Section 2.3.1) and they might be interrupted by other spontaneous movements such as kicking.

For this, it would be interesting to detect these movements, in order to filter away frames where it is difficult to assess the GMs. It should be noted that kicking/punching is considered as movements with high velocity and it is thus natural to use these features for classification. Figure 8.4 illustrates the velocity of the right foot for two infants. It is easy to observe that one infant is kicking its legs a lot more, compared to the other infant.

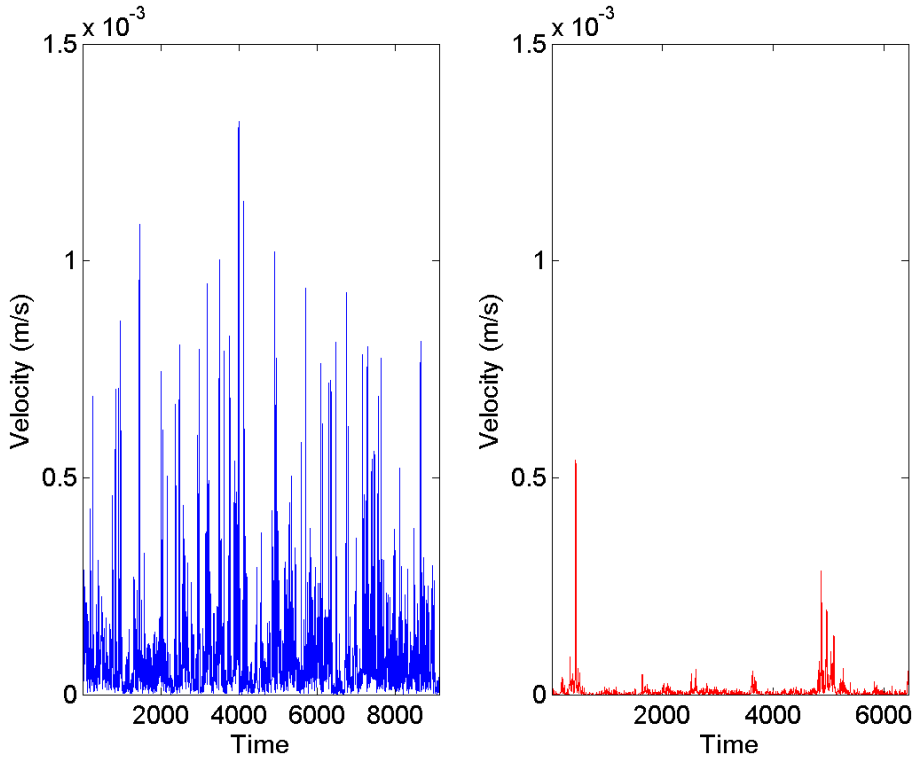


Figure 8.4: The velocity of the foot for two infants is visualized. One infant is kicking with its legs and the other infant is quite calm.

Even though kicking movements might be relatively easy to detect, simply based on the velocity, the problem is solved using machine learning. In order to train a classifier for detecting the spontaneous movements, such as kicking, a number of frames of different infants has been annotated with either spontaneous or calm movements. Using features such as maximum and total velocity/acceleration within a neighborhood of the frames, a number of different classifiers has been trained. Both k-nearest neighbor, classification tree and support vector machine classifiers [27] has been trained, which resulted in similar results, all of them resulting in an accuracy above 90%. Figure 8.5 shows the segmentation of the

data for one infant, together with the ground truth annotations. It is seen that the two plots look rather similar.

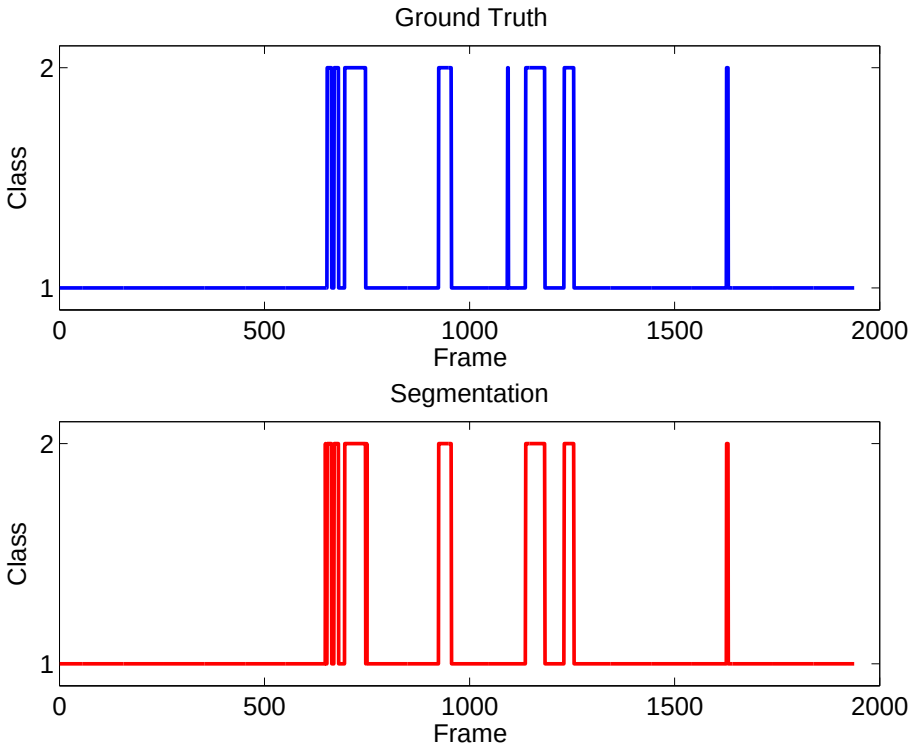


Figure 8.5: Based on the kicking-classifier, the frames has been classified with or without kicking movements.

8.4 GM Detection

As mentioned in Section 2.3.1, the Fidgety Movements (FMs) are considered in this study, due to their higher predictive power over Writhing Movements (WMs). The FMs are described as small movements in the joints. However, there is no implicit mathematical formulation of these movements, even though existing studies focus on training features related to normal or abnormal measurements. The different features used in the existing studies are described below.

- [4]: Features based on pixel-changes between frames in Two-dimensional (2D) video recordings. Features were generated from the amount of changes as well as the position, velocity and acceleration of the centroid of motion.
- [78]: Using optical flow, the position of body parts are tracked in time, resulting in trajectories of the different body parts. Features are generated based on frequency and wavelet analysis of these trajectories.
- [45]: Based on tracking markers in Three-dimensional (3D), features extracted from the trajectories of these markers are used for classification. The features are based on velocity, acceleration and different statistical measurements from before mentioned distributions.

It is clear that the common choice of features are based on velocities and accelerations as well as features describing smoothness. In order to identify frames containing movements related to GMs and more specifically FMs, a number of features are generated for each of the body parts. Until now, the features have been based on one or two frames, resulting in features describing stationary configurations or velocity and acceleration. In the following, some of the features are based on a window approach, where the features are generated from one frame and its neighboring frames. Given a window of size w, the features for the frame f is generated from data extracted from the frames; $f-w, f-w+1, \ldots, f, \ldots, f+w-1, f+w$. The size of w can be tailored for the specific features, but if nothing else is written, $w = 15$, resulting in features generated from approximately one second of data. For a respective frame f, a segment of different trajectories is extracted. Four kinds of trajectories are considered, namely the trajectory of the body part's start and end point, as well as the global and local direction vectors. The global direction vector is the normalized direction vector between the start and end point. For each of the trajectories, a number of features are extracted. This is e.g. maximum/minimum/total/mean of the motion quantities velocity/acceleration/curvature. Furthermore, features describing the distance between two points in the segment, such as start- and end-point, is also included. All these features are extracted for each limb and for each frame. Based on these features, the goal is to identify frames containing FMs. In order to do so, data have been annotated with FMs. Two types of annotations are used, namely those where the FMs occur globally, as well as some where the FMs occur in individual body parts. Based on a leave one out approach, a classification tree is trained based on annotated positive frames. In this work it is not possible to calculate the ROC-statistics, as frames are labeled as positive, while non-positive frames does not necessarily indicate a negative annotation. Only statistics based on true positives and false negatives can be calculated. Furthermore, as the annotation process does not give a perfect segmentation of the frames, the annotations might be slightly shifted with respect to the ground truth movements. Because of this, the performance is

not measured using a one-to-one comparison between the annotations and the classification. For each annotation, the time-difference to the closest positive classifications is measured. A summary of the distances is visualized as a histogram plot in Figure 8.6. The amount of annotations which has been classified within a time deviation of 0.2 seconds dominates the results. Some false negative classifications can be seen as well, which can be seen as the occurrences of distance up to 2 seconds.

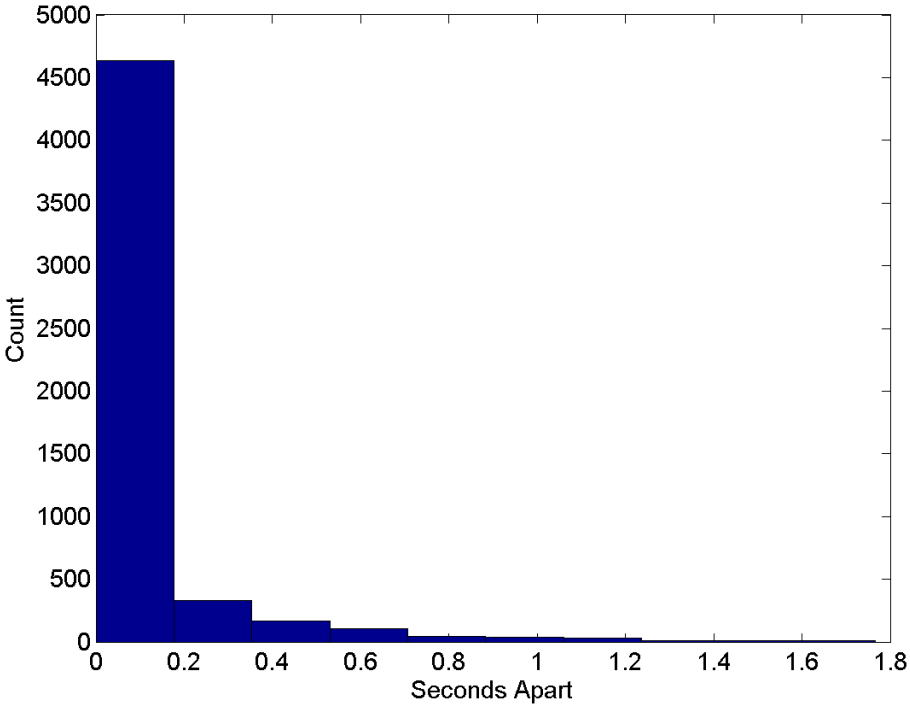


Figure 8.6: The distances between the annotated frames and the closest classified frames are summarized in a histogram plot. Most frames have been classified correctly within 0-2 seconds, but other frames have wrongly been classified as false negatives.

It should be noted that the analysis is not completely fair, in the sense that no ground truth negatives are available in the data. An all-positive classifier would thus result in a perfect score in terms of the measure in Figure 8.6. Another way to illustrate that the classification is correct, is to visualize the results as a function of the infants' age. From the literature (Section 2.3.1) it is known that the FMs are most apparent between three to five months Corrected Age (CA). It is thus expected that the classification will show this behavior. Figure 8.7

illustrates the CA-FM plot. The plot shows the normalized amount of FMs detected using the trained classifier. It is seen that there is a trend of increased FMs for infants in the age of 12-20 weeks. Some infants in this age group have less FMs, but this is acceptable as the FMs should not appear all the time during the recording. It should be noted that none of the infants have been diagnosed with Cerebral Palsy (CP) and future studies should validate the analysis using data from infants with abnormal/absent FMs.

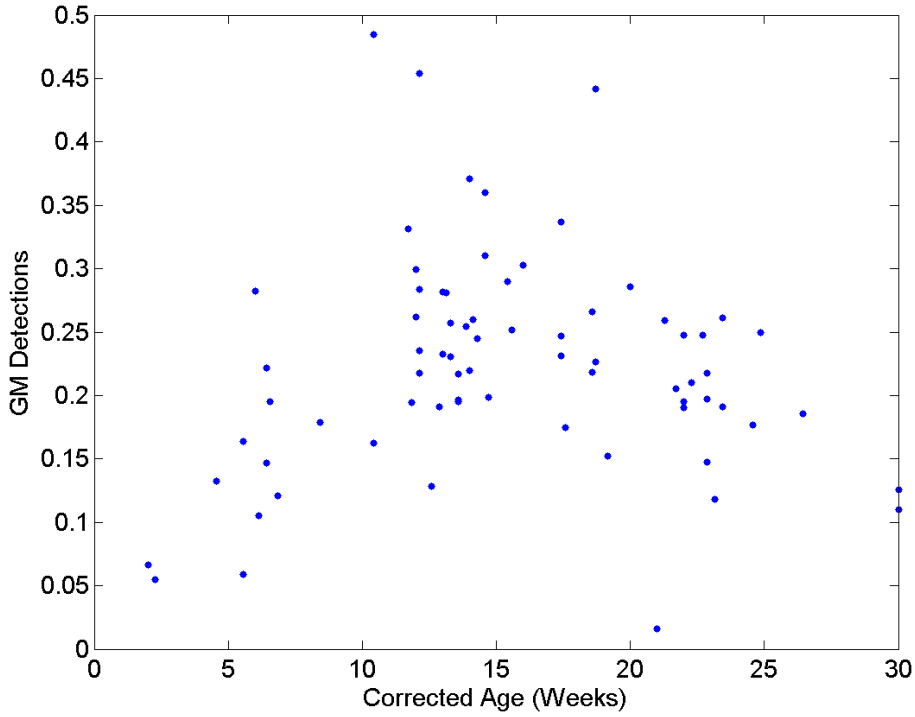


Figure 8.7: The normalized amount of classified FM frames for the different infants are shown. See the trend that the amount peaks in the age of 12-20 weeks.

8.5 Concluding Remarks

Using the results from the motion tracking, different methods were used for classifying the infant's movements, such as detecting if the infant was able to meet certain motor milestones. The presented examples are only used for demonstrating the possibilities of the motion tracking system and other quantities could be

extracted as well. As for the classification of GMs, a classifier was trained on annotations of FMs. Even though the results are promising, further validation is required with respect to acquiring true abnormal/absent movements.

Chapter 9

Conclusion

9.1 Conclusion

In this work, we presented several methods for estimating the pose of infants and tracking their limbs. This has been done using data from a low-cost depth sensor, which is able to capture both color and depth information of a scene. The motion tracking is done based on two approaches.

The first approach was able to detect the position and identify different body parts in the data. This was done based on the simple assumptions that the infant could be reduced to an object consisting of outgoing parts. Even though this is a rough assumption, our results showed that the method was able to correctly identify and locate the different body parts. However, as shown in the thesis, the method has troubles estimating the correct pose, when the infant joins multiple body parts.

In order to make the motion tracking more robust, a model-based approach was developed. This approach models the surface of the infant, as well as the different constraints of the human body, thus resulting in a more anatomical correct representation of the infant. This second method greatly improved the pose estimation and the tracking results.

Different features were extracted based on the results from the motion tracking system. This included individual body part specific features, such as joint velocities and accelerations. Other features were based on interactions between different body parts such as joint-to-joint distances and angles. We also presented a novel method for estimating scene flow from human data, estimating a dense 3D flow field of the infants surface. The denser representation of movements can be used to assess small movements not necessarily described by the infant body model.

Based on the extracted features, we presented several examples for assessing the infant's movements and behavior. This includes different milestones, such as the infant's ability to reach for the feet, roll from side to side and coordinate the limbs in both sides of the body. Other quantities such as kicking and crying are also considered, based on machine learning and image analysis. The different classification scenarios presented in this work are only examples of potential applications. However, the potential is almost limitless as long as the type of movements can be quantified by the results from the motion tracking.

Using expert annotations of fidgety movements, a decision tree was trained to detect these movements. The classification shows promising results, but it should be noted that further validation is needed in order to make an automatic system for diagnosing cerebral palsy. Currently, none of the included infants have been diagnosed with cerebral palsy, even though some of them was born preterm. Follow-ups on the infants might lead to new conclusions regarding the analysis and classification.

9.2 Future Considerations

The presented motion tracking system is based on a general description of the infant body without any prior information, except general anatomical constraints for the modeled joints. However, based on the huge amount of data acquired during this study, this can be further utilized for improving the motion tracking, both with respect to robustness and speed. Currently, the motion tracking is done offline and real-time motion tracking is not possible on the highest data resolution. However, learning the appearance of infants could potentially result in a device similar to the Microsoft Kinect, which is able to do real-time markerless motion tracking of adults, without the need for any kind of calibration. The appearance prior could even be used for doing 3D motion tracking of infants, only based on color-images. Close collaboration with other groups working on early identification of cerebral palsy could thus be a reality, in spite of different data modalities, as video recordings are often included in these studies.

Other considerations are the ever increasingly accessibility of 3D measuring devices. Previously, 3D measurements was only accessible in the fields of aerial imaging, civil engineering, entertainment, etc. Nowadays, a smartphone is enough to create detailed 3D models of various objects and can actually make a 3D scan of a person's face. Some laptops even have a build in depth sensor, which is able to track a person's face or hands. It is only a matter of time before most smartphones probably comes with depth sensor as well. With this technology, motion tracking of infants could be done at the infant's home. A simple mount for a smartphone could be placed above the infant's bed and used for tracking the infants movements. The data could then be transferred to a therapist, whom made the final assessment of the infant's motor skills.

Appendix A

Appendix

A.1 Levenberg-Marquardt

The formula for calculating the optimal update step in the Levenberg-Marquardt (LM) method is derived in the following.

The goal is to find the vector δ_θ that minimizes the sum of squared errors in Equation A.1

$$\begin{aligned}
\delta_\theta^* &= \arg\min_{\delta_\theta} \sum_i (\mathbf{x}_i - \mathbf{f}_i(\theta) - (\mathbf{J}\delta_\theta)_i)^2 && \text{(A.1)}\\
&= \arg\min_{\delta_\theta} \sum_i (\mathbf{r}_i - (\mathbf{J}\delta_\theta)_i)^2 && \text{(A.2)}\\
&= \arg\min_{\delta_\theta} (\mathbf{r} - \mathbf{J}\delta_\theta)^T (\mathbf{r} - \mathbf{J}\delta_\theta) && \text{(A.3)}
\end{aligned}$$

Differentiating Equation A.3 with respect to the update step, results in a simple linear least squares problem which can be solved directly.

$$\begin{aligned}
\frac{\partial (\mathbf{r} - \mathbf{J}\delta_\theta)^T (\mathbf{r} - \mathbf{J}\delta_\theta)}{\partial \delta_\theta} &= \frac{\partial (\mathbf{r}^T\mathbf{r} - \mathbf{r}^T\mathbf{J}\delta_\theta - (\mathbf{J}\delta_\theta)^T\mathbf{r} + (\mathbf{J}\delta_\theta)^T\mathbf{J}\delta_\theta}{\partial \delta_\theta} && \text{(A.4)}\\
&= \frac{\partial (\mathbf{r}^T\mathbf{r} - \mathbf{r}^T\mathbf{J}\delta_\theta - \mathbf{r}^T\mathbf{J}\delta_\theta + \delta_\theta^T\mathbf{J}^T\mathbf{J}\delta_\theta}{\partial \delta_\theta} && \text{(A.5)}\\
&= 0 - \mathbf{r}^T\mathbf{J} - \mathbf{r}^T\mathbf{J} + 2\delta_\theta^T\mathbf{J}^T\mathbf{J} && \text{(A.6)}\\
&= -2\mathbf{r}^T\mathbf{J} + 2\delta_\theta^T\mathbf{J}^T\mathbf{J} && \text{(A.7)}
\end{aligned}$$

Setting this equal to zero and solving for δ_θ gives the final solution:

$$\begin{aligned}
-2\mathbf{r}^T\mathbf{J} + 2\delta_\theta^T\mathbf{J}^T\mathbf{J} &= \mathbf{0}^T && \text{(A.8)}\\
-2\mathbf{J}^T\mathbf{r} + 2\mathbf{J}^T\mathbf{J}\delta_\theta &= \mathbf{0} && \text{(A.9)}\\
\mathbf{J}^T\mathbf{J}\delta_\theta &= \mathbf{J}^T\mathbf{r} && \text{(A.10)}\\
\delta_\theta &= (\mathbf{J}^T\mathbf{J})^{-1}\mathbf{J}^T\mathbf{r} && \text{(A.11)}
\end{aligned}$$

A.2 Description of Graphical User Interface

During the project, a graphical user interface has been developed for applying the pose estimation and motion tracking. The code is written in C#(C Sharp) using various libraries for 3D rendering, Graphics Processing Unit(GPU) utilization and accessing the Kinect data. The application consists of three windows, namely Main Window, Sideview and ControlPanel.

A.2.1 Main Window

The **Main Window** is the primary window for visualization. This window renders the 3D data and the 3D model. Using keyboard inputs, the camera can be moved freely in the 3D space, making it possible to observe the rendered scene from different orientations.

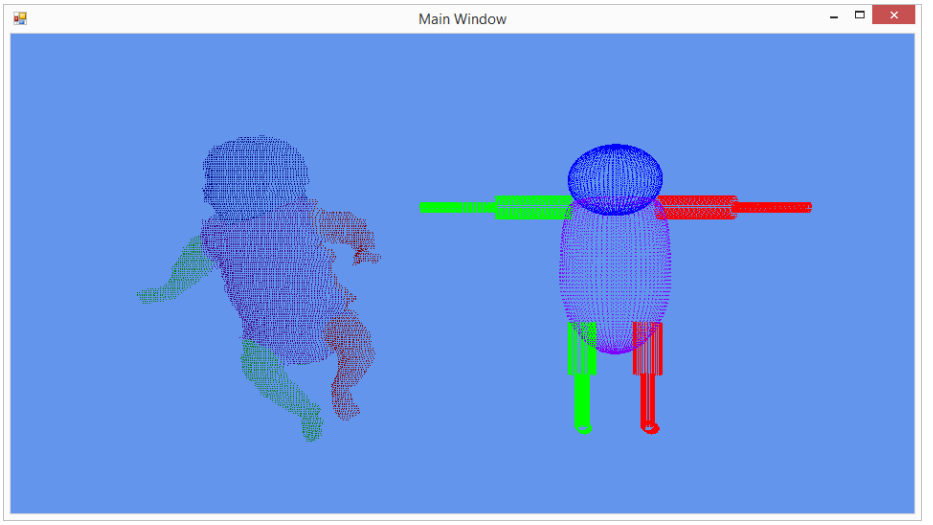


Figure A.1: The Main Window shows the full scene and can be used for observing the data, as well as the 3D model.

A.2.2 Sideview

The **Sideview** window is only for assisting the user in observing the 3D rendering. The sideview shows the rendered scene from the side, relative to the viewing direction of the camera in the Main Window. The field of view for this window is relative small, allowing the user to observe slices of the 3D rendered scene.

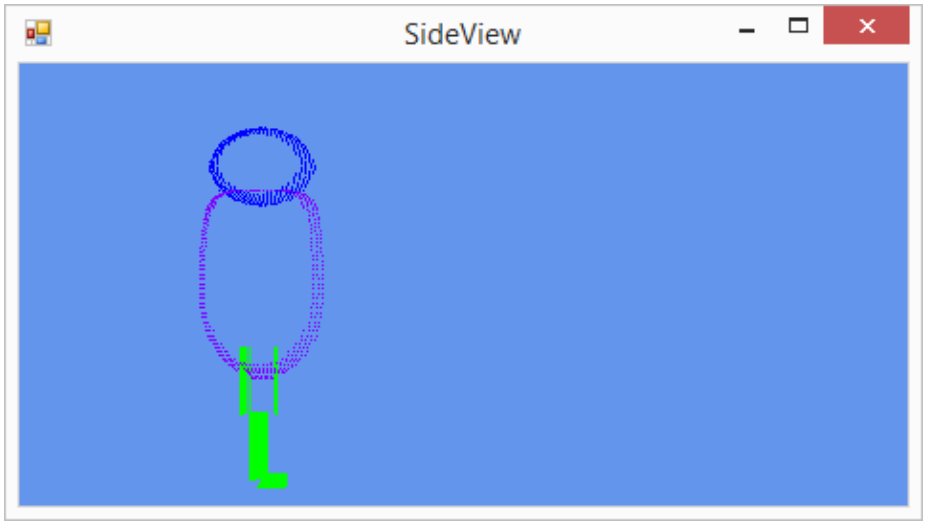


Figure A.2: The Sideview is used for observing the 3D scene from the side.

A.2.3 ControlPanel

The **ControlPanel** is the window that is used to manage the analysis as well as extracting relevant data. The ControlPanel can be divided into several regions, based on functionality. These regions will be summarized below.

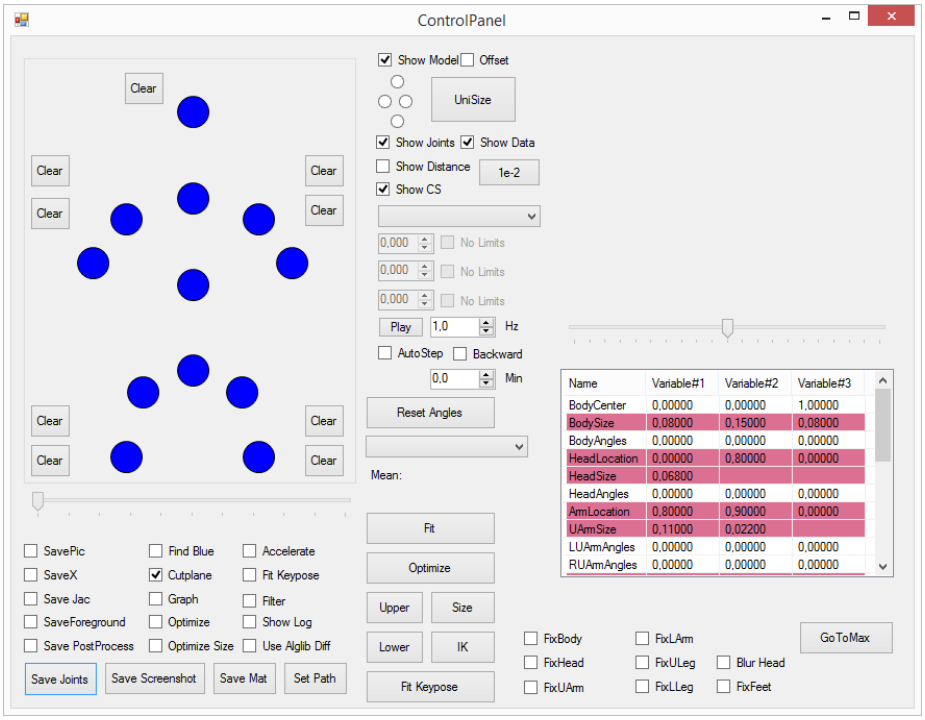
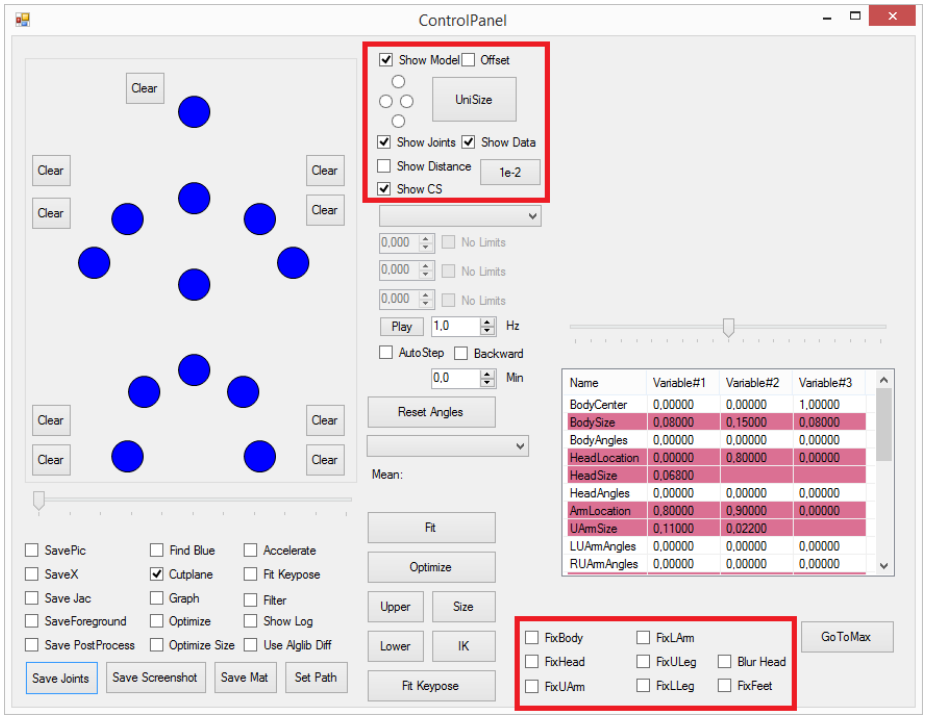


Figure A.3: The ControlPanel window contains all the functionally for managing the pose estimation and motion tracking.

A.2.3.1 Visualization

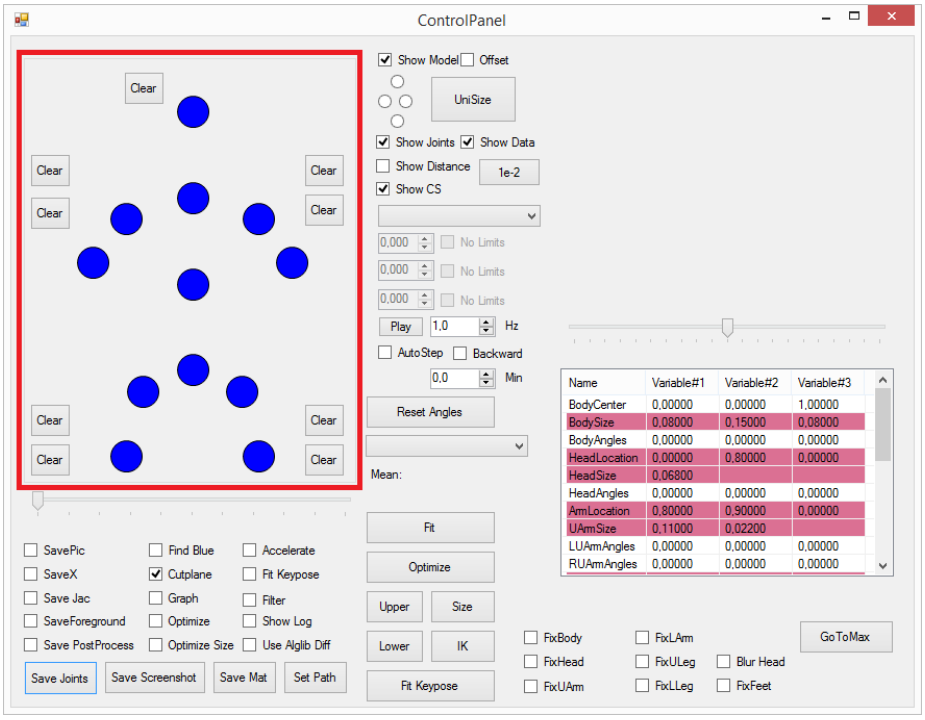
- Upper Region: Different configurations can be used for the 3D rendered scene. The visibility of the 3D model and data can be toggled and a set of assisting objects can be shown as well. The model joints can be visualized by local coordinate systems that visualize their position and global orientation. As it might be beneficial to rotate the data, this can be done as well, where the four cardinal directions can be used for the up-direction. Furthermore, an offset can be added to the model, in order to observe the data and the model without any overlap. A fixed number of box-objects can be shown as well, that naturally follow the joint positions of the 3D model. Their functionality is explained in Section A.2.3.2.

- Lower Region: For visualization purposes, the different body parts can be fixed on the 3D model. The reason can e.g. be to remove movements in the upper arms/legs, in order to only observe the movement in the lower legs/arms/feet. In addition, the input data can be anonymized by removing the color in the head-region. This is under the assumption that the pose is known.



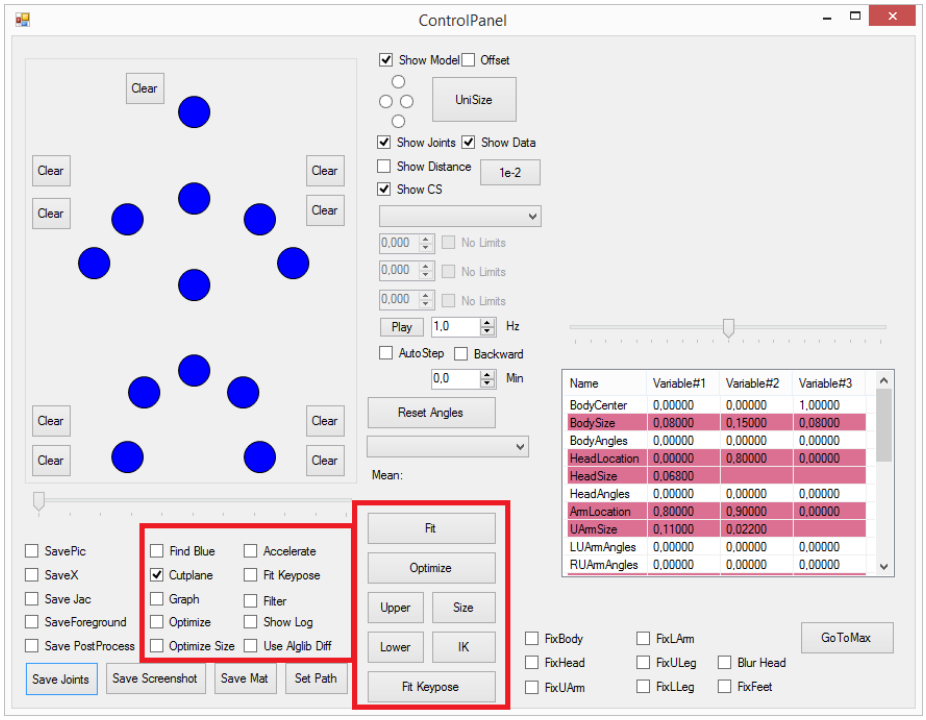
A.2.3.2 Joint Manipulation

A set of joint-objects can be visualized on top of the 3D model. These joints follow the 3D model. The objects can be selected either by clicking on the objects in the rendered scene or by selecting the respective blue joint in the ControlPanel window. When a joint is selected, it can be moved around in the 3D rendered scene. This can both be done using the keyboard or, if available, a leap motion sensor can be used. Using the leap motion sensor, the selected point follows the movements of the user's hand, which gives a naturally way of positioning the joints. The position of the joints can later be used for fitting the 3D model using inverse kinematics.



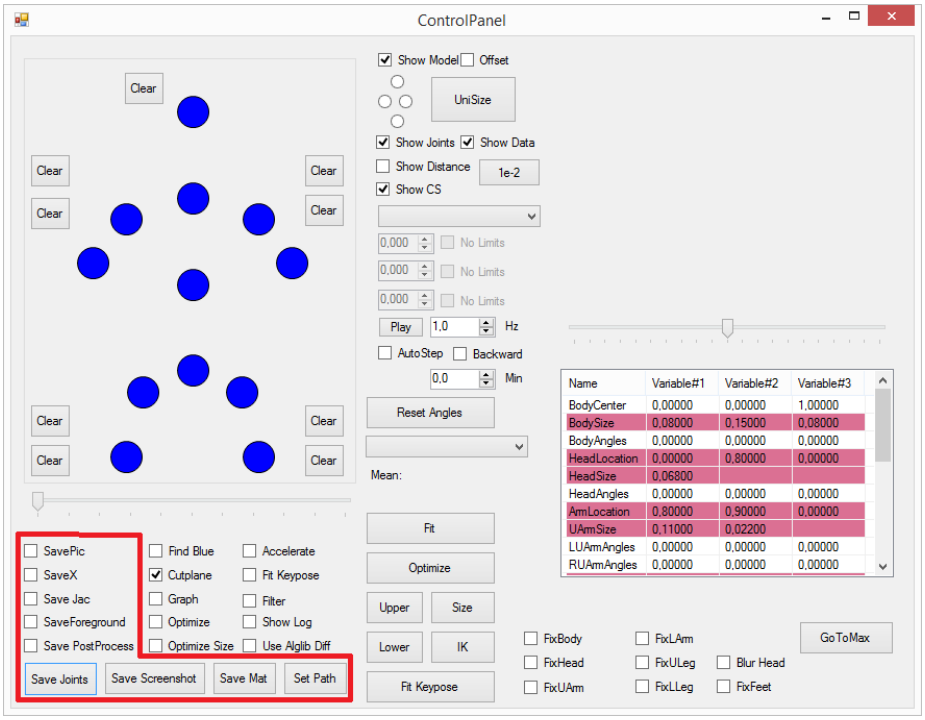
A.2.3.3 Model Fitting

- Right Region: Different approaches can be used for fitting the 3D model to the data. This can either be based on the graph-based approach combined with inverse kinematics, the common-poses approach or the direct model-based approach. The user can freely choose to apply the methods in any order, but some methods are better suited to be used prior to others. Furthermore, the size parameters can be optimized with respect to the data. This can be done globally or only to the lower/upper body regions.
- Left Region: When a new frame is obtained either from the Kinect or from the harddrive, the different fitting methods can be applied automatically. Furthermore, additional settings can be toggled on/off, including printing log-information in a command window, filtering the parameters from frame to frame and preprocessing the data.



A.2.3.4 Data Extraction

A number of methods can be used for extracting data. This can e.g. be taking a screenshot of the rendered scene, saving the current pose parameters and joint positions or save the data in a Matlab file.

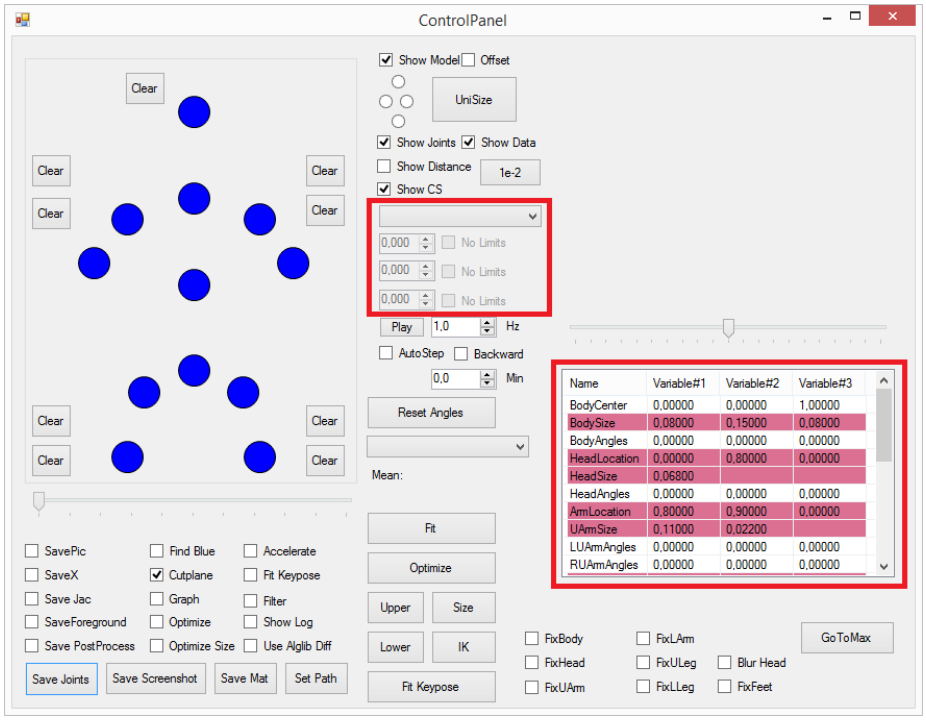


A.2.3.5 Variable Management

The pose parameters can be observed and modified in the ControlPanel:

- Left Region: Using a Drop-down menu, the different parameters can be chosen. This includes both orientation, position and size parameters. Next, the values of the parameters can be changed. Changing one parameter affects the 3D model immediately.

- Right Region: A table gives an overview of the parameters and their values. The color of the rows indicate if a value is fixed or not. The table supports copy/paste in the sense that the values for the respective parameters can be copied from one frame to another using the hotkeys for copy and paste, respectively.



A.2.4 Easy Postprocessing and Data Extraction

For easy analysis of the data, a Matlab class was implemented as well that is able to load the results from the motion tracking system. This contains the functionality for easy visualization of the body pose (see Figure A.4). Both angular and spatial representations are directly available to the user. Furthermore, the respective depth and color data for each frame can be accessed and further processed. This includes estimation of the scene flow and segmentation of body specific regions in the depth and color images.

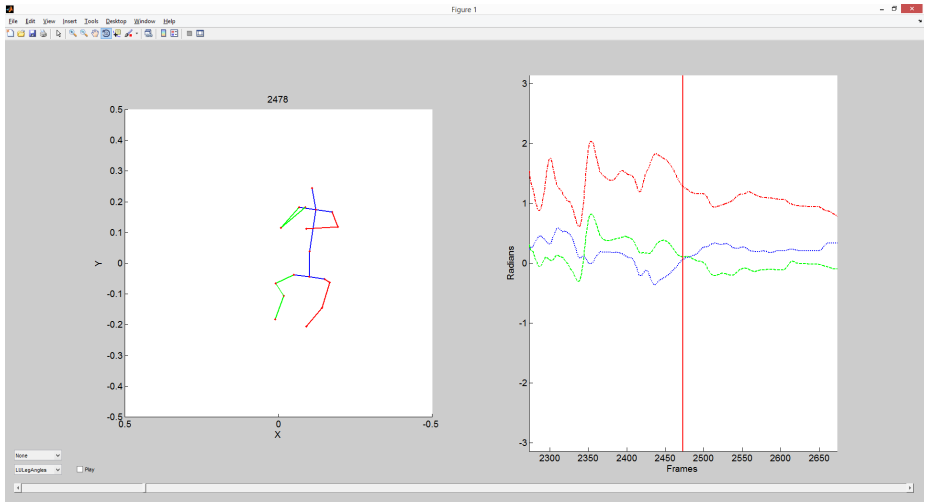


Figure A.4: The Matlab window for visualizing the skeleton pose and motion data

Part II

Contributions

Appendix A

Body-Part Tracking of Infants

Accepted for poster presentation at the International Conference of Pattern Recognition 2014 in Stockholm.

2014 22nd International Conference on Pattern Recognition

Body Part Tracking of Infants

Mikkel Damgaard Olsen*, Anna Herskind†, Jens Bo Nielsen†‡ and Rasmus R. Paulsen*

*Department of Applied Mathematics and Computer Science
Technical University of Denmark, Richard Petersens Plads 324
DK-2800 Kongens Lyngby, Denmark

†Department of Neuroscience and Pharmacology
and
‡Department of Nutrition, Exercise and Sport
University of Copenhagen, Blegdamsvej 3
DK-2200 Copenhagen, Denmark

***Abstract*—Motion tracking is a widely used technique to analyze and measure adult human movement. However, these methods cannot be transferred directly to motion tracking of infants due to the big differences in the underlying human model. However, motion tracking of infants can be used for automatic analysis of infant development and might be able to tell something about possible motor disabilities such as cerebral palsy. In this paper, we address markerless 3D body part detection of infants using a widely available depth sensor and discuss some of the major challenges that arise. We present a method to detect and identify a set of the anatomical extremities and the results are evaluated based on manually annotated 3D positions.**

I. Introduction

Tracking research has attracted considerable interest in recent years, but only few studies exist in which the methods have been used for motion tracking in infants even though the techniques may be of benefit in the evaluation of infant development. However, some work exists, regarding motion tracking of infants. In [1], the authors use an optical motion system, where reflective markers are attached to the infant's limbs and multiple infra-red cameras are used to reconstruct the 3D location of the markers in space. From this, a set of parameters is extracted and used for early detection of spasticity due to cerebral palsy. In [2], [3] a non-optical approach is used to extract the motion parameters, where 6 sensors are attached to the infants' wrists, ankles, chest and head. The sensors give temporal information about position and orientation. In [4], the authors propose a new, optical flow-based method for quantifying the motion of infants with neonatal seizures, based on an overall quantitative measure of movement between successive frames. The optical flow-based approach is also used in [5], where the authors use color images as input to an optical flow-algorithm in order to track the position of the infants' arms and legs. Here, the authors manually initializes the position of the different body parts and adjust the positions during the tracking, in order to improve robustness of the method.

In this work, we look at an automatic system for detection of relevant body parts on infants without the need for marker-based systems or initialization of points. This is done using an affordable and easy-to-use depth sensor, Microsoft Kinect, which has revolutionized research within the field of low cost motion tracking. In a broader perspective, the goal is to use this work for motion analysis of infants to be able to improve the diagnostics of different diseases, namely cerebral palsy (CP). CP is the most common motor disability among children, affecting 2-2.5 out of 1000 infants [6]. It is caused by an injury of the fetal or infant brain, and the physical impairment is in many cases accompanied by disturbances of cognition and perception [7]. Among several others, preterm birth and birth asphyxia are associated with an increased risk of CP, but frequently a clear underlying pathology is not found [8], [9]. During infancy symptoms are often subtle, but early warning signs include failure to meet motor milestones such as crawling and walking [10]. Due to the lack of unequivocal symptoms, in current practice, most children with CP are not diagnosed until the age of 2 years [6]. However, studies have shown that the movement patterns of infants are influenced by CP already in the months before and after birth [11]. For fetuses, the movements can be observed and analyzed using ultrasound, while the post-birth studies are often based on analyzing videos of the infants' so-called general movements, which can be observed until the age of 5 months (corrected w.r.t term). By observing and identifying the motion patterns, cerebral palsy may thus be suspected at a much earlier age. Early identification of infants at risk of CP leads to the possibility of early intervention, which may improve the development of the infants' motor and cognitive skills. However, due to the time consuming procedure of analyzing the motion patterns, it is unrealistic to examine all infants born at risk of CP. The method presented in this work will thus try to move closer to an automatic system. The idea is that the system should be standard equipment, e.g. located at outpatient clinics, used for analyzing movement patterns of premature infants. Because of this we cannot use a complex system that requires hours of preparation and calibration, but it should be a sort of plug-and-play system. In order to do this automatically, a number of problems must be considered. The question is, given the recorded data, how the system can find, model and track the movement of the infant over time. One answer to this is by use of vision-based motion tracking.

A. Related Work

Vision-based motion tracking is not a new research area, but as new techniques for capturing data are developed, new

1051-4651/14 $31.00 © 2014 IEEE
DOI 10.1109/ICPR.2014.377

ways of acquiring the data used for motion tracking arise. A number of surveys exists [12], [13] that summarize some of the work done within the field of motion tracking. However, as the data used for motion tracking is obtained using the Microsoft Kinect Sensor, the motion tracking that comes with the sensor is a good place to start. [14] describes the tracking algorithm implemented in the Kinect sensor. Here, the authors use a relatively simple depth feature and a learned classifier, based on random forests, to do fast recognition of body parts, given a single depth image. However, as the tracking is based on a learned classifier, the tracking does not work for all human figures, such as infants, even though the classifier has been trained on a big dataset. This tracking approach has also been used in other papers such as [15], [16] with good results. The approach of learning the appearance of the human figure, is also utilized in [17], where point sets in a database are aligned to the observed point cloud. From the best pose candidate, a rough estimate of the underlying human skeleton can be determined and further refinement can be applied. Other work exists, which does not rely on learned appearance, but on the anatomical limitations to the human body. In [18], [19] it is assumed that the human body contains a number of limbs (arms, legs and head). By representing the observed 3D data as a graph, the outer body parts can be found as points furthest away from the center of the body. From this, a rough estimate of the human skeleton can be found and used for further refinement. In [20] the method is extended, where optical flow is used to distinguish between overlapping body parts. A less a priori based method is described in [21], in which a skeleton model is fitted to a point cloud, using a genetic algorithm. Here a predefined skeleton model is evolved iteratively through random mutations and crossovers, in order to fit the model to the observed data. The only a priori information needed, is the structure of the underlying skeleton, i.e. the number of limbs and their relative connections. Inspired by existing methods, we will in the following describe the pipeline used to do body part detection on infants, only using the raw data obtained from a low cost depth sensor.

II. Methods

The data used in this work, are depth and color images acquired with the Kinect sensor from Microsoft. The depth images have been recorded at a resolution of 480×640 pixels and the same resolution is used for the color images. However, it is possible to get better color images, at the cost of frames per second. The participating infants are 3-6 months of age. For each infant 15-30 minutes of data have been recorded, while the infants have been lying on a flat surface e.g. a mattress or a blanket (see Figure 1). It should be noted that the pictures and data are used and published with respect to an agreement signed by the participating families. The Kinect has been positioned above the infant using a tripod for ordinary cameras. Using the depth images, a 3D point cloud representation of the infant can be generated and used to detect and locate the limbs of the infant. In order to make this easier, the fact that the infant lies on a flat surface is used to differentiate between foreground (most likely the infant) and background, by fitting a least squares plane to the surface. This is simply done by solving the linear system:

$$a(\mathbf{x} - x_c) + b(\mathbf{y} - y_c) + c(\mathbf{z} - z_c) = \mathbf{d}, \qquad (1)$$

Fig. 1. Color image of one of the recorded infants. The infant wears an easily recognizable (blue) bodystocking and lies on a white blanket.

where; $(\mathbf{x}, \mathbf{y}, \mathbf{z})$ are the observed 3D points of the underlying surface, (x_c, y_c, z_c) is a 3D point on the plane, and (a, b, c) are the elements for the normalvector of the plane. $\mathbf{d}$ defines the signed distances to the plane, but is equal to the zero vector during the fitting process. Giving the solution, the signed distance from every 3D point to the plane can be calculated and this is used to discard points behind the plane ($\mathbf{d} < 0$). In order to remove small deviations of the underlying surface, a threshold is used instead of the value zero. It is assumed that the viewing direction of the camera is nearly perpendicular to the flat surface and thus, the normal is corrected to point towards the camera. In addition, the infant wears an easily recognizable bodystocking, which is used to locate the baby, but this will be explained later.

A. Skeleton Modeling

Inspired by previous work [22], [23], we have chosen to model the skeleton as a hierarchical model, with the root starting from the center of the body. The identification of the body center is based on the center of mass of a set of classified pixels. As the infants wear a colored bodystocking during the recording, this can be recognized and tracked in the data. The reason for choosing the body center as the root is that as the infant is lying on its back, the location of the body center should be the most static part of the infant and more movement should be seen in the outer limbs. A visual representation of the model can be seen in Figure 2, where the lines represent limbs and the nodes represent joints. Some pseudo joints have been made, such as chest and stomach, in order to simplify the overall structure of the model. Three colors are used to indicate the joints'/limbs' relative location with respect to the root node. Red is left, green is right and purple is neutral.

B. Locating Extremities

In order to locate the anatomical extremities, i.e. hands, feet and the head, the method described in [18], [19], [20] is used. It is assumed that the anatomical extremities are located farthest away from the center of the body. By farthest, we mean with respect to geodesic distances, i.e. distances on the surface of the body. In order to estimate the geodesic distances, the 3D data is represented as a graph, where every pixel is connected to its closest neighbors. As the 3D data originate from a depth map, the neighborhood of each 3D point follows directly from the neighborhood in the depth image. The weights of the edges in the graph are calculated as the Euclidean distances between

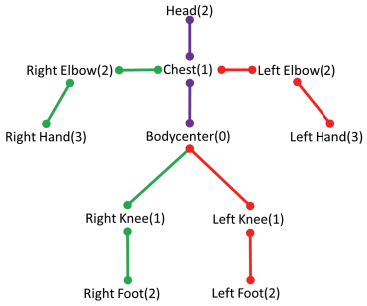


Fig. 2. Hierarchical model shown as a skeleton with the front towards the reader, where nodes represent joints and lines represent limbs connecting the joints. The numbers simply relate to the level in the hierarchical representation.

the respective 3D points. Once the graph has been constructed, we use Dijkstra's shortest path algorithm [24] to calculate the distance from the body center node to all other nodes in the graph. The node with the maximum distance is chosen as the first anatomical extremity. In order to locate additional extremities, the solution is not to choose the node with the second largest distance, as this will most likely result in a node that is close to the first extremity node. A number of solutions to this issue have been proposed. In [18] a zero-cost edge is added from the body center to the extremity node and the shortest path algorithm is repeated. In [19], the authors set the latest found extremity node as the source node and update the graph distances. In [20] the shortest path algorithm is executed once and a predefined threshold is used to segment the body into a number of possible anatomical regions. For each region the node with the largest geodesic distance is chosen as the anatomical extremity. If the previously described methods are to be used for motion tracking of infants, two problems arise. In [20], the predefined threshold is based on the size of the person, making infant tracking difficult, as infants vary significantly in size. Furthermore, there is no guarantee that the threshold will segment the body into the correct number of regions. In [18], [19] an anatomical extremity may be identified between the center of mass and a previous extremity, which is undesirable. In order to solve these two problems, we use an alternative approach, which will be explained in the following.

1) As an initial step, a graph is constructed from the depth image, where each pixel is a node and each node's neighborhood is defined by the respective pixel's neighborhood in the depth image. The edge-weights are based on the Euclidean distances between the respective pixels in 3D.
2) The source node is chosen as the body center of the infant. This body center is identified based on a segmentation of the bodystocking that the infant wears, where simple color analysis is used to identify the pixels belonging to the bodystocking.
3) The shortest path from the source node to every node in the graph is calculated by applying the shortest path algorithm once. The information of the traveled path is stored, in order to be able to backtrack every pixel to the body center.
4) The node with the maximum geodesic distance is denoted as the next anatomical extremity.
5) As in [18], [19] the next step is to prevent that the same extremities are found in the following passes of the shortest path algorithm. However, instead of only setting the distance of the extremity node to zero, all nodes on the path from the body center node to the extremity node are changed. I.e. if a node on a hand is detected as an anatomical extremity, both distances of the hand node as well as all nodes on the upper and lower arm are set equal to zero. By doing this, we do not only penalize the hand, but the entire arm is penalized and should thus not be chosen as the path of a new extremity.
6) Based on the updated distances, the shortest path algorithm is applied again to get a new geodesic distance map and the next anatomical extremity is detected.

The described procedure is repeated until the desired number of anatomical extremities is obtained. In the original Dijkstra's algorithm, the nodes are visited based on a sorted list of distances. This requires a sorting of the nodes, but ensures that each node is only visited once. As the data originate from a depth image, we do not use the original Dijkstra's shortest path algorithm, but a slightly modified version. We use a priority queue to store the nodes that are to be visited. Using a simple first-in, first-out procedure, the node-distances are updated in the order they are found. The missing sorting step, results in the possibility of visiting nodes multiple times, but due to the image structure of the nodes, this will rarely happen and the used approach has experimentally been shown to be faster than using the sorting step. However, this is not a general conclusion, but only a conclusion for the data used in this study.

Once the extremities are located, the next step is to classify the extremities as one out of a set of predefined anatomical body parts. In [18], [19] a local descriptor is constructed, which describes the shape of the extremity and this descriptor is compared to descriptors in a database. As an alternative, our approach does not need a learned database of patches. Instead we assume that if the orientation of the body is known, the path from the body center to the anatomical extremities can be used as a descriptor of the different body parts. The different extremities are thus identified based on their spatial location and the orientation of the geodesic path from the body center. This assumption is reasonable as the infants lie in a controlled environment and in the more general case, humans will most likely have their heads upwards, meaning that this assumption can also be used for more general motion tracking tasks. Once the anatomical extremities have been identified, sub-extremities such as elbows, knees and chest are located based on fractional distances from the body center to their respective parent-extremities. In Figure 3, a color frame of a recording is seen with the visualization of the skeleton, where the positions of the detected body parts have been connected, based on the skeleton model described earlier. It is observed that the sub-extremities such as elbows and knees are located slightly off the correct positions. As described, these joints are found at an fractional distance from the body center to the parent-extremities, following the shortest path. This results in

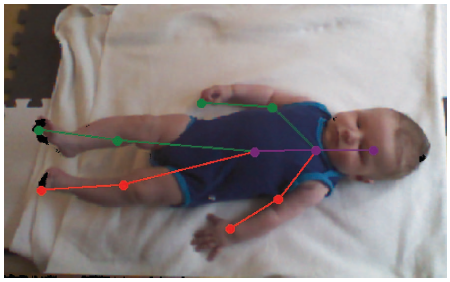


Fig. 3. The color image of an infant, overlaid with the detected skeleton, to show that the tracking is able to obtain a valid estimate of the body parts.

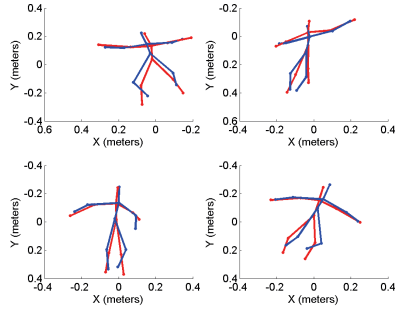


Fig. 4. Examples of frames, where the tracking did find a good estimate of the underlying skeleton. The blue skeleton is the ground truth observation and the red skeleton is the result from the body detection.

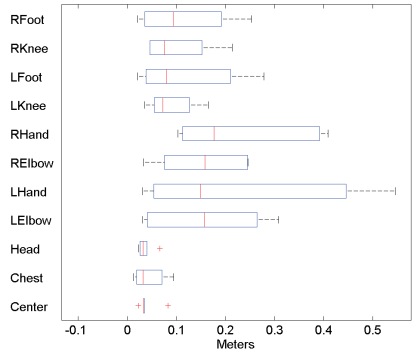


Fig. 5. A boxplot showing the tracking errors for each individual body part. The errors are based on all available frames that contained ground truth information.

the joints lying at the border of the respective limbs, rather than on the medial axis. For future versions of the method, this should be avoided. One solution is to weight nodes based on the distance to the border, e.g. using a distance map on the depth image. Another solution is to use the current model to find an initial skeleton model and refine the model based on the point cloud.

III. Results

In order to test the body part detection, 6 Kinect recordings of infants have been used for testing. Each infant has been recorded for 15-30 minutes. In order to be able to compare the position of the detected body parts, reference data has been generated, by manually annotating the positions of the individual joints in 3D, on a limited number of frames. The annotations have been done using a tool implemented in parallel with this study. The tool is able to show real time point clouds from the Kinect sensor and simple 3D primitives. Here, the primitives represent the reference points and these can freely be moved around in the 3D world. To ease this process in an intuitive way, the Leap sensor from LeapMotion[1] is used to directly manipulate the position of the landmarks in three directions, based on the position of the user's hand. It should be noted that the annotations have been placed in the interior and not on the surface of the 3D data, which knowingly will result in deviations from the estimated joint positions. The reason for using the interior positions, is that this is closer to the anatomical positions of the joints. Based on the previously described hierarchical skeleton model, the estimated joints can be connected and illustrated in 3D. However, we have chosen to ignore the depth/Z data in the illustrations, because this makes it easier to illustrate the model. In Figure 4, the results from the presented method can be observed. We see that the method correctly locates and identifies the different body parts. Two skeletons can be observed in the figures, where a blue skeleton shows the visualization of the ground truth annotations and the red skeleton is tracking results. We see that the two skeletons fit each other and that the method works as expected. The residuals of the individual body parts can be seen in Figure 5, where the boxplot gives an overview of the Euclidean distances between the position of the detected body parts and the ground truth positions. We see that the method is especially good at finding the body center and the head where the method gives very consistent estimations. In fact the residuals of the body center and head positions are mainly due to the differences in depth, caused by the fact that the ground truth points are positioned in the interior of the point cloud, while the tracked points are found on the surface. The boxplot also shows that the outer body parts, such as the hands and feet, not surprisingly have the biggest errors, as the motion of these body parts, dominates in the overall motion of the infant. In addition these body parts tend to make complex connections in relation to the graph and this also influences the final tracking results. To show some of the complex cases mentioned above, Figure 6 illustrates the estimated skeletons on top of the RGB color frames. For the two cases to the left, the underlying blanket (background) has changed significantly and has thus become a part of the foreground data, which obviously results in a wrong detection of the infant. In the two remaining cases to the right, the infants lie in a posture that makes wrong graph-connections between the different body parts and thus the detection and identification cannot be done correctly. Future work will focus on these kinds of problems, where some might be solved by requiring a more controlled

[1] http://www.leapmotion.com

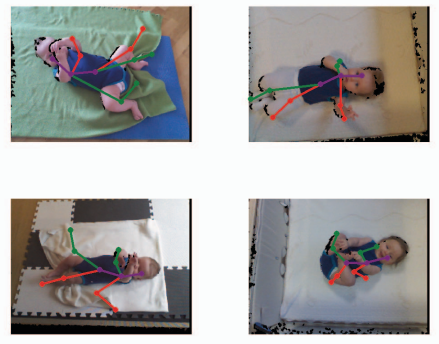


Fig. 6. Examples of frames, where the tracking failed to locate the correct positions of the body parts, due to changes in the background model (left) and the posture of the infant (right).

environment, while others should be solved by changing the detection procedure.

IV. Conclusion

We have presented a method that is able to locate, identify and track the different body parts of an infant using only data obtained from the Microsoft Kinect sensor. No a priori information or learned classifier/database is necessary to do the tracking and the method does not require any time consuming set up procedure or calibration. The method is based on representing a depth image as a graph, and from this detect and locate anatomical extremities using Dijkstra's shortest path algorithm. Based on the infant's orientation the method is able to identify the extremities as one out of a predefined set of body parts. We have evaluated the results based on manual annotations of the correct positions in the data and found out that the method is able to robustly identify and detect the body center and head of the infants. The method can also find other extremities, such as hands and feet, when the data does not contain complex postures, such as crossing of arms or feet. The system only requires a depth sensor and an ordinary camera tripod and thus, the system is ideal as a simple plug-and-play application. As a proof of concept, the presented work shows that a simple and automatic system for analyzing movements of infants is realistic and that this knowledge brings us closer to an automatic tool for diagnosing infants with motion disabilities such as cerebral palsy. For future work, we will focus on the more complex postures of the infants and how to make the background subtraction more robust to changes in time.

Acknowledgements

The authors would like to thank all of the infants and their families for participating in this project, and the organization APA, for helping make contact with the families.

References

[1] L. Meinecke, N. Breitbach-Faller, C. Bartz, R. Damen, G. Rau, and C. Disselhorst-Klug, "Movement analysis in the early detection of newborns at risk for developing spasticity due to infantile cerebral palsy." *Human movement science*, vol. 25, no. 2, pp. 125–144, Apr. 2006.

[2] A. Berg, "Modellbasert klassifisering av spedbarns bevegelser," 2008.

[3] P. Rahmanpour, "Features for movement based prediction of cerebral palsy," 2009.

[4] N. B. Karayiannis, B. Varughese, G. Tao, J. D. F. Jr., M. S. Wise, and E. M. Mizrahi, "Quantifying motion in video recordings of neonatal seizures by regularized optical flow methods," *IEEE Transactions on Image Processing*, vol. 14, no. 7, pp. 890–903, 2005.

[5] A. Stahl, C. Schellewald, O. Stavdahl, O. M. Aamo, L. Adde, and H. Kirkerod, "An optical flow-based method to predict infantile cerebral palsy," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 20, no. 4, pp. 605–614, 2012.

[6] K. Himmelmann, "Epidemiology of cerebral palsy," *Handbook of Clinical Neurology*, vol. 111, pp. 163–167, 2013.

[7] M. Bax, M. Goldstein, P. Rosenbaum, A. Leviton, N. Paneth, B. Dan, B. Jacobsson, and D. Damiano, "Proposed definition and classification of cerebral palsy, april 2005," *Developmental Medicine & Child Neurology*, vol. null, pp. 571–576, 8 2005.

[8] S. Goldsmith, N. Badawi, E. Blair, D. Taitz, J. Keogh, and S. McIntyre, "A systematic review of risk factors for cerebral palsy in children born at term in developed countries," *Developmental Medicine and Child Neurology*, vol. 55, no. 6, pp. 499–508, 2013.

[9] S. McIntyre, C. Morgan, K. Walker, and I. Novak, "Cerebral palsy-don't delay." *Dev Disabil Res Rev*, vol. 17, no. 2, pp. 114–29, 2011.

[10] N. Murphy and T. Such-Neibar, *Cerebral Palsy Diagnosis and Management: The State of the Art*, ser. Current problems in pediatric and adolescent health care, 2003, pp. 149–69.

[11] C. Einspieler, H. Prechtl, A. Bos, F. Ferrari, and G. Cioni, *Prechtl's Method on the Qualitative Assessment of General Movements in Preterm, Term and Young Infants*, ser. Clinics in Developmental Medicine. Wiley, 2008.

[12] T. B. Moeslund and E. Granum, "A survey of computer vision-based human motion capture," *Comput. Vis. Image Underst.*, vol. 81, no. 3, pp. 231–268, Mar. 2001.

[13] T. B. Moeslund, A. Hilton, and V. Krüger, "A survey of advances in vision-based human motion capture and analysis," *Comput. Vis. Image Underst.*, vol. 104, no. 2, pp. 90–126, Nov. 2006.

[14] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 1297–1304.

[15] C. Keskin, F. Kirac, Y. Kara, and L. Akarun, "Real time hand pose estimation using depth sensors," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, 2011, pp. 1228–1234.

[16] S. Escalera, P. Radeva, D. Dimov, M. Reyes, A. Marinov, A. Hernandez-Vela, and N. Zlateva, "Graph cuts optimization for multi-limb human segmentation in depth maps," *2012 IEEE Conference on Computer Vision and Pattern Recognition*, vol. 0, pp. 726–732, 2012.

[17] M. Ye, X. Wang, R. Yang, L. Ren, and M. Pollefeys, "Accurate 3d pose estimation from a single depth image," in *Proceedings of the 2011 International Conference on Computer Vision*, ser. ICCV '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 731–738.

[18] C. Plagemann, V. Ganapathi, D. Koller, and S. Thrun, "Real-time identification and localization of body parts from depth images," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 2010, pp. 3108–3113.

[19] A. Baak, M. Müller, G. Bharaj, H.-P. Seidel, and C. Theobalt, "A data-driven approach for real-time full body pose reconstruction from a depth camera," in *IEEE 13th International Conference on Computer Vision (ICCV)*. IEEE, Nov. 2011, pp. 1092–1099.

[20] L. A. Schwarz, A. Mkhitaryan, D. Mateus, and N. Navab, "Estimating human 3d pose from time-of-flight images based on geodesic distances and optical flow." in *FG*. IEEE, 2011, pp. 700–706.

[21] D. L. Ly, A. Saxena, and H. Lipson, "Pose estimation from a single depth image for arbitrary kinematic skeletons," *Computing Research Repository*, 2011.

[22] C. Mnier, E. Boyer, and B. Raffin, "3d skeleton-based body pose recovery." in *3DPVT*. IEEE Computer Society, 2006, pp. 389–396.

[23] S. Shen, M. Tong, H. Deng, Y. Liu, X. Wu, K. Wakabayashi, and H. Koike, "Model based human motion tracking using probability evolutionary algorithm." *Pattern Recognition Letters*, vol. 29, no. 13, pp. 1877–1886, 2008.

[24] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms, Third Edition*, 3rd ed. The MIT Press, 2009.

Appendix B

Model-Based Motion Tracking of Infants

Accepted for poster presentation at the European Conference on Computer Vision 2014 in Zurich.

Model-Based Motion Tracking of Infants

Mikkel Damgaard Olsen[1(✉)], Anna Herskind[2], Jens Bo Nielsen[2,3], and Rasmus Reinhold Paulsen[1]

[1] Department of Applied Mathematics and Computer Science, Technical University of Denmark, Kongens Lyngby, Denmark
mdol@dtu.dk
[2] Department of Neuroscience and Pharmacology, University of Copenhagen, Copenhagen, Denmark
[3] Department of Nutrition, Exercise and Sport, University of Copenhagen, Copenhagen, Denmark

Abstract. Even though motion tracking is a widely used technique to analyze and measure human movements, only a few studies focus on motion tracking of infants. In recent years, a number of studies have emerged focusing on analyzing the motion pattern of infants, using computer vision. Most of these studies are based on 2D images, but few are based on 3D information. In this paper, we present a model-based approach for tracking infants in 3D. The study extends a novel study on graph-based motion tracking of infants and we show that the extension improves the tracking results. A 3D model is constructed that resembles the body surface of an infant, where the model is based on simple geometric shapes and a hierarchical skeleton model.

Keywords: 3D model fitting · Infant pose estimation · Markerless motion tracking · Depth images

1 Introduction

Motion tracking of humans has attracted considerable interest in recent years, but only few studies exist in which the methods have been used for motion tracking of infants. However, the techniques can be of great benefit in the evaluation of infant development, as they can quantify the movements of infants and might be able to improve the diagnostics of different motor related diseases, namely cerebral palsy (CP). CP is the most common motor disability among children, affecting 2-2.5 out of 1000 infants [1]. It is caused by an injury to the fetal or infant brain, and the physical impairment is in many cases accompanied by disturbances of cognition and perception [2]. Among several others, preterm birth and birth asphyxia are associated with an increased risk of CP, but frequently a clear underlying pathology is not found [3,4]. During infancy symptoms are often subtle, but early warning signs include failure to meet motor milestones such as crawling and walking [5]. Due to the lack of unequivocal symptoms, in current practice, most children with CP are not diagnosed until the age

L. Agapito et al. (Eds.): ECCV 2014 Workshops, Part III, LNCS 8927, pp. 673–685, 2015.
DOI: 10.1007/978-3-319-16199-0_47

of 2 years [1]. However, studies have shown that the movement patterns of infants are influenced by CP already in the months before and after birth [6]. In fetuses, the movements can be observed and analyzed using ultrasound, while the post-birth studies are often based on analyzing videos of the infants' so-called general movements, which can be observed until the age of 5 months (corrected w.r.t term). By observing and identifying the motion patterns, cerebral palsy may thus be suspected at a much earlier age. Early identification of infants at risk of CP leads to the possibility of early intervention, which may improve the development of the infants' motor and cognitive skills. However, due to the time consuming procedure of analyzing the motion patterns, it is unrealistic to manually examine all infants born at risk of CP. The method presented in this work will thus try to move closer to an automatic system. The idea is that the system should be standard equipment, e.g. located at outpatient clinics, used for analyzing movement patterns of prematurely born infants. Because of this we cannot use a complex system that requires hours of preparation and calibration, but it should be a sort of plug-and-play system. As mentioned, studies within this field of interest are limited, but some work exists on motion tracking of infants. In [7,8] the authors use 6 sensors attached to the infants' wrists, ankles, chest and head. The sensors give temporal information about position and orientation. In [9], the authors use an optical motion system, where reflective markers are attached to the infant's limbs and multiple infra-red cameras are used to reconstruct the 3D location of the markers in space. From this, a set of parameters is extracted and used for early detection of spasticity due to cerebral palsy. In [10], the authors propose a new, optical flow-based method for quantifying the motion of infants with neonatal seizures, based on an overall quantitative measure of pixel-differences between successive frames. The optical flow-based approach is also used in [11], where the authors use color images as input to an optical flow-algorithm in order to track the position of the infants' hands and feet. Here, the authors manually initialize the position of the different body parts and adjust the positions during the tracking, in order to improve robustness of the method. In [12] the authors describe a method for tracking the 3D positions of anatomical extremities(hands, feet, head) and sub-extremities(elbow, knee) of infants based on a graph-based method equivalent to the approach in [13–15]. Based on the the work of [12], the method is extended with a model-based approach as in [16,17]. This both improves the body part localization and the tracking of the infants movements over time. The data is obtained using an affordable and easy-to-use depth sensor, Microsoft Kinect, which has revolutionized research within the field of low cost motion tracking.

2 Methods

The data used in this work, are depth and color images acquired with the Kinect sensor from Microsoft. The depth images have been recorded at a resolution of 480×640 pixels and the same resolution is used for the color images. As far as the authors' knowledge, no benchmark database of dense 3D data of infants exist

Fig. 1. Color image of one of the recorded infants. The infant wears an easily recognizable (blue) bodystocking and lies on a white blanket.

and a non-public database is used, which has been created simultaneous with this study. The participating infants are 3-8 months of age. For each infant 15-30 minutes of data have been recorded, while the infants have been laying on a flat surface e.g. a mattress or a blanket (see Figure 1). It should be noted that the pictures and data are used and published with respect to an agreement signed by the participating families. The Kinect has been positioned above the infant using a tripod for ordinary cameras. Using the depth images, a 3D point cloud representation of the infant and its surroundings can be generated. In order to make tracking easier, the fact that the infant lies on a flat surface is used to differentiate between foreground (most likely the infant) and background, by fitting a least squares plane to the surface. This is simply done by solving the linear system:

$$a(\mathbf{x} - x_c) + b(\mathbf{y} - y_c) + c(\mathbf{z} - z_c) = \mathbf{d}, \tag{1}$$

where; $(\mathbf{x}, \mathbf{y}, \mathbf{z})$ are the observed 3D points of the underlying surface, (x_c, y_c, z_c) is a 3D point on the plane, and (a, b, c) are the elements for the normalvector of the plane. $\mathbf{d}$ defines the signed distances to the plane, but is set to zero during the fitting process. It is assumed that the viewing direction of the camera is nearly perpendicular to the flat surface and thus, the normal is corrected to point towards the camera. Given the estimated plane parameters, the signed distance from every 3D point to the plane can be calculated and this is used to discard points behind the plane ($\mathbf{d} < 0$). In order to remove small deviations of the underlying surface, a non-zero value is used as threshold. In addition, the infant wears an easily recognizable bodystocking, which is used to locate the baby, using color-based pixel classification.

2.1 Body Model

In this work, we use a 3D model to describe the surface of the human body. The model is constructed from a predefined number of geometric shapes that are connected based on the underlying skeleton. In order to measure the distance from the body model to the observed data, a "point to shape"-distance function is defined for each shape [16,17]. Currently the geometric shapes are:

– Cylinder: Used for describing elongated body parts, such as arms, legs and feet. The distance between a 3D point and the cylinder can be found analytically, by projecting the point onto the medial axis of the cylinder and taking the thickness/radius of the cylinder into account. The distance function for at 3D point $\mathbf{p}$ is thus defined as:

$$d = \begin{cases} \|\mathbf{p} - \mathbf{a}\| & \text{if } \lambda \leq 0 \\ \|\mathbf{p} - \mathbf{b}\| & \text{if } \lambda \geq 1 \\ \|\mathbf{p} - (\mathbf{a} + \lambda\,(\mathbf{b} - \mathbf{a}))\,\| & \text{otherwise} \end{cases}, \tag{2}$$

where, $\mathbf{a}$ and $\mathbf{b}$ are the start- and endpoints of the cylinder and

$$\lambda = \frac{(\mathbf{p} - \mathbf{a}) \cdot (\mathbf{b} - \mathbf{a})}{\|\mathbf{b} - \mathbf{a}\|^2} \tag{3}$$

is the normalized length of the vector $(\mathbf{p} - \mathbf{a})$ projected onto the line $(\mathbf{b} - \mathbf{a})$. It should be noted, that this distance is only for calculating the distance from a 3D point to the medial axis of a cylinder and the radius of the cylinder should be included in order to calculate the distance to the surface. Moreover, the described distance function, represents the distance to a rounded cylinder, when the radius/thickness is included.

– Sphere: Used to describe the head of the infant. The distance is easily computed, as the distance between the center point and the 3D point. Again, the radius of the body part should be included, in order to measure the distance to the surface, rather than the distance to the center.
– Ellipsoid/Superquadratic: The torso/stomach is modeled as a combination of two superquadratics. The upper part is modeled as a round-cornered box, in order to describe the box-like shape of the shoulders, while the lower part is modeled as a simple ellipsoid. In this case, no closed form solution exists for calculating the exact euclidean distance. Instead two differnet approaches has been used, which approximates the true distance:
  1. One solution is to use a numerical method, as explained in [18], which minimizes the distance from the 3D point to a point on the surface of the ellipsoid. The method is not generalized to superquadratics, but it is also possible to use iterative methods for approximating the distance for the upper bodypart [19].
  2. Another solution is to create a distancemap of the bodypart. Once the distancemap is created, the distance from the superquadratics to a 3D point can be approximated, by mapping the 3D point to the distancemap coordinates.

Inspired by previous work [20,21], we have chosen to model the skeleton as a hierarchical model, with the root starting from the center of the body. The identification of the body center is based on the center of mass of a set of classified pixels. As the infants wear a colored bodystocking during the recording, this can be recognized and tracked in the data. The fact that the infant lies on its back, is the reason the body center is chosen as the root joint. This is because the

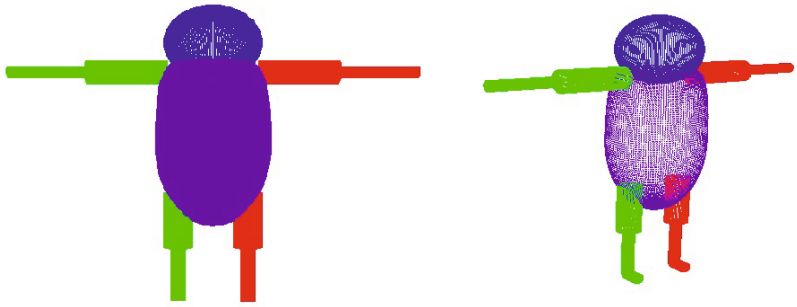


Fig. 2. Visual 2D (left) and 3D (right) representation of the model used in this study. The colors are simply to differentiate the bodyparts. Red and green are used to differentiate between left and right bodyparts, respectively.

location of the body center should be the most static part of the infant and more movement should be seen in the outer limbs. The articulated model can be seen in Figure 2, where colors are used to differentiate between left(red) and right(green) bodyparts.

In relation to the hierarchical connection between the bodyparts, the geometric shapes are oriented and positioned, using simple rotations and translations. However, instead of modeling the rotations in the global coordinate system, with euler angles, axis-angle representations are used, where rotations are limited to local coordinate systems, as the local coordinate system of each joint, changes based on the parent joints. The definition of axis-angle rotations is described widely in the literature [22,23] where the Rodriquez rotation formula can be used to construct a rotation-matrix given a axis-angle representation. Given an axis of rotation ω and a rotation angle θ, the rotation matrix can calculated as:

$$\mathbf{R} = \mathbf{I}_3 + \hat{\omega} \sin\theta + \hat{\omega}^2 (1 - \cos\theta) . \tag{4}$$

$\mathbf{I}_3$ is the 3×3 identity matrix and the $\hat{}$ operator constructs the skew symmetric matrix of the vector ω.

2.2 Fitting the Model

In order to fit the model to the observed data, the Levenberg Marquardt method is used, to iteratively refine the body parameters. The state vector and objective function will thus be defined in the following. An overview of the state parameters used in this study are listed in Table 1. Only the *Stomach* bodypart has a spatial parameter, which controls the global position of the model, while the position of the remaining bodyparts are constrained based on the hierachical model and the size parameters of the body.

The size parameters define the size and relative location of the bodyparts and are listed in Table 2. The size parameters are not part of the optimization but are either given prior to the optimization or estimated during an initialization

Table 1. Overview of the orientation parameters used for each bodypart. As can be seen, only one bodypart (*Stomach*) has a spatial parameter.

Bodypart	**Parameters**
Stomach	Rotation, Position
Head	Rotation
Left Upper Arm	Rotation
Right Upper Arm	Rotation
Left Lower Arm	Rotation
Right Lower Arm	Rotation
Left Upper Leg	Rotation
Right Upper Leg	Rotation
Left Lower Leg	Rotation
Right Lower Leg	Rotation
Left Foot	Rotation
Right Foot	Rotation

Table 2. Overview of the size parameters used for each bodypart. These parameters are not part of the optimization, but are used during the creation of the 3D model.

Bodypart	**Size**	**Location**
Stomach	Extension for the three axis	Global
Head	Radius	Relative to Stomach
Left Upper Arm	Radius + Length	Relative to Stomach
Right Upper Arm	Radius + Length	Relative to Stomach
Left Lower Arm	Radius + Length	Relative to Left Upper Arm
Right Lower Arm	Radius + Length	Relative to Right Upper Arm
Left Upper Leg	Radius + Length	Relative to Stomach
Right Upper Leg	Radius + Length	Relative to Stomach
Left Lower Leg	Radius + Length	Relative to Left Upper Leg
Right Lower Leg	Radius + Length	Relative to Right Upper Leg
Left Foot	Radius + Length	Relative to Left Lower Leg
Right Foot	Radius + Length	Relative to Right Lower Leg

step. It should be noted that symmetry is utilized and it is thus assumed that symmetric bodyparts are identical, with respect to size and relative location.

Once the state parameters are defined, the next step is to optimize on these parameters using the Levenberg-Marquardt optimization scheme, in order to fit the 3D model to the observed data. By concatenating the state parameters in a state vector $\mathbf{x}$, the optimization can be written as:

$$\min_{\mathbf{x}\in\mathbb{R}} \sum_{i=1}^{N} \|\mathbf{p}_i - \mathbf{c}\left(\mathbf{p}_i, \mathbf{x}\right)\|, \tag{5}$$

where $\mathbf{c}\left(\mathbf{p}_i, \mathbf{x}\right)$ calculates the closest 3D point on the model, given the 3D data point $\mathbf{p}_i$ and the state vector. An extension to the above minimization, is that

the state vector $\mathbf{x}$ is constrained, based on the anatomical properties of the human body joints. One requirement for the Levenberg-Marquardt algorithm, is an initial starting guess. In this study, a good estimate of the starting guess is found using an existing method for detecting and locating anatomical extremities, based on graph theory [12]. Here the anatomical extremities such as head, hands and feet are located by assuming that these points are furthest away from the bodycenter, when the distance measure is based on geodesic distances over the body surface. The distance is estimated by representing the surface as a graph, where neighboring 3D points are connected by nodes. This approach is able to locate and identify both the extremities and sub-extremities such as elbows and knees. The described method is able to give an estimate on the spatial location of the extremities. In order to obtain the respective state vector, an inverse kinematics method is used.

The total body-modeling method is summarized in the following, where input is the data obtained from the Kinect sensor.

1. Apply background subtraction in order to segment the infant from the background/underlying surface.
2. Define Body Parameters either using fixed parameters or during an initialization step.
3. Use graph-based method to obtain an estimate on the location of the anatomical extremities.
4. Apply an inverse kinematics algorithm in order to obtain the state parameters, given the location of the extremities(end effectors).
5. Refine the state parameters, in order to minimize the distance between the 3D model and the observed data.

The above described approach, estimates the orientation parameters of a single frame. However, by using the optimized parameters of one frame, as starting guess for the successive frame, the human body can be tracked in time.

3 Results

In order to test and evaluate the described method, Kinect recordings of 7 infants' movements have been obtained, where each infant has been recorded for 15-30 minutes. As no ground truth data is available, a various number of frames have been manually annotated for each infant. The frames have been selected such that they cover a wide variety of poses. In Figure 3, the results from the presented method can be observed. The method is able to correctly locate and identify the different body parts and the joint angles can be extracted directly from the respective state vector. It should be noted that an offset of the 3D model has been used, to better illustrate the estimated pose.

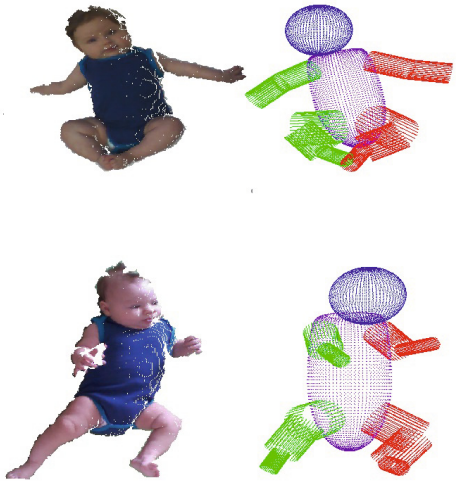


Fig. 3. Examples of the (translated) model fitted to the observed data. The point clouds are colored for visualization purposes, but only the 3D information is used during the optimization process.

In the following, the two methods (graph-based vs. model-based) are compared. The Euclidean distance between the manually positioned 3D points and the estimated joint locations are used to evaluate the tracking approach. In Figure 4 and Figure 5, the mean and standard deviation of the residuals are shown, respectively. The reader should notice that the *Stomach* and *Chest* residuals does not differ significantly, as these locations are found almost equivalently. However, the localization of the remaining joints has improved significantly, both with respect to mean and standard deviation.

The results indicate how robust the method is to locate the different body parts. As mentioned earlier, the technique can easily be extended to motion tracking, instead of human body detection. In Figure 6 the Euclidean distances between successive frames can be observed, for four different joints. Is is noteworthy to see that the graph-based tracking contains a lot of peaks/noise, while the model-based tracking gives a more smooth tracking. The reason for this, is that the model-based approach is less sensitive to deviations in data, compared to the graph-based method.

Based on the tracking results of one of the infants, Figure 7 shows the angles of the upper arms and the thighs, during a time period of 45 seconds. It is observed that the right upper arm is less active, compared to the remaining body parts.

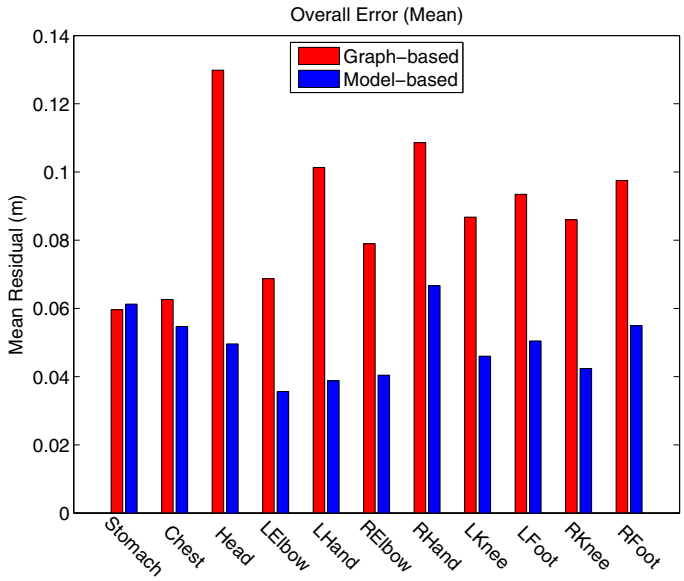


Fig. 4. The mean of the residuals listed for each body part in the model. Both the results from the graph-based approach (*red*) and the extended method (*blue*) are visualized.

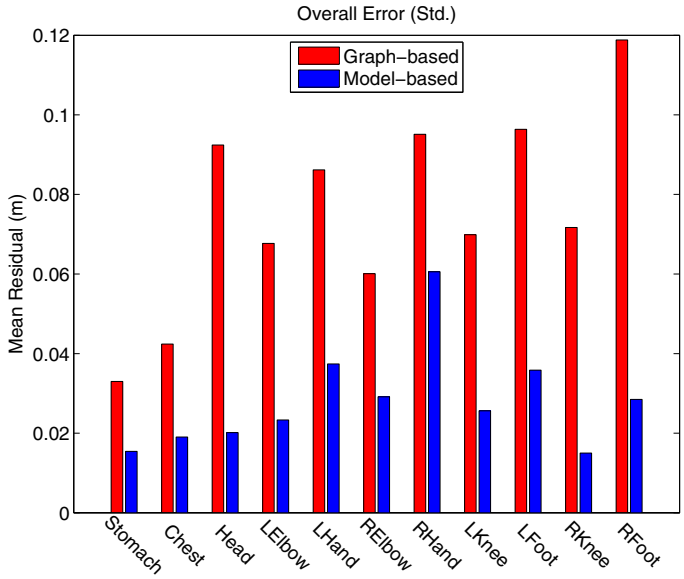


Fig. 5. The standard deviation(*right*) of the residuals listed for each body part in the model. Both the results from the graph-based approach (*red*) and the extended method (*blue*) are visualized.

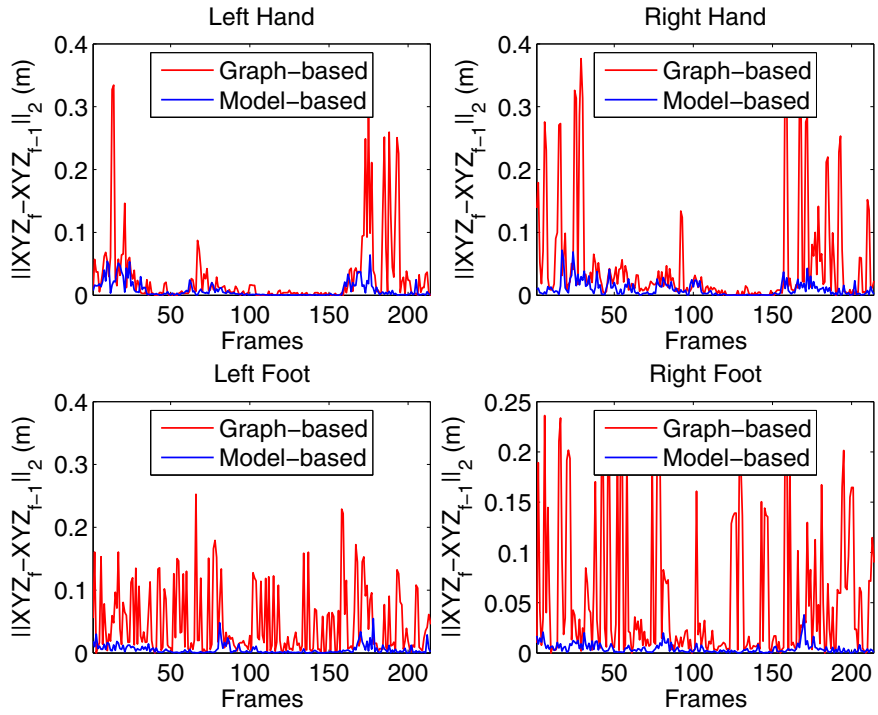


Fig. 6. Comparison of tracking results for the two methods. For each frame the Euclidean distance to the previous joint-location is calculated and visualized.

In Figure 8 the positions of the elbows and knees are visualized, which gives a better visualization of how the right elbow is much more passive, compared to the other bodyparts. This overall motion-plot might help doctors and physiotherapists to detect abnormal motion behaviors and plan the training according to these results.

Even though the study shows promising results, the tracking sometimes gets stuck in local minima, due to certain postures of the infant. One problem is e.g. when the infant rolls from supine to prone position. Here the method is unable to recover the correct orientation of some body parts and this error propagates through the tracking. This might be solved by enforcing additional temporal filtering on the body parameters. Another solution would be to create a pose library as in [16], where a number of candidate poses are evaluated and tested against the estimated pose.

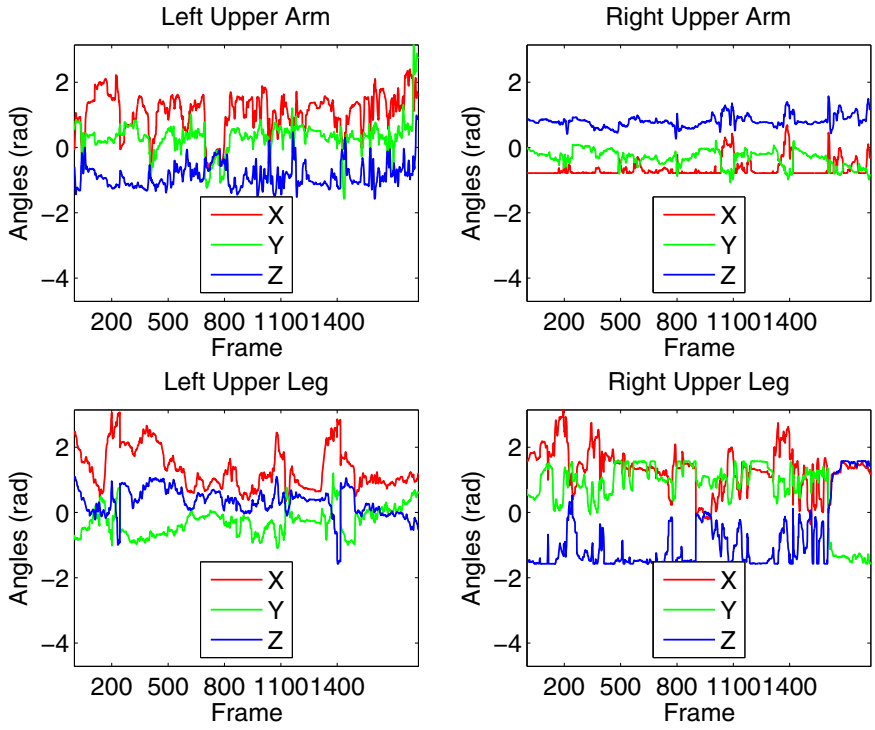


Fig. 7. The angles with respect to the local x-, y- and z-axis, for four bodyparts. It is noteworthy that the right upper arm is less active, compared to the three other bodyparts.

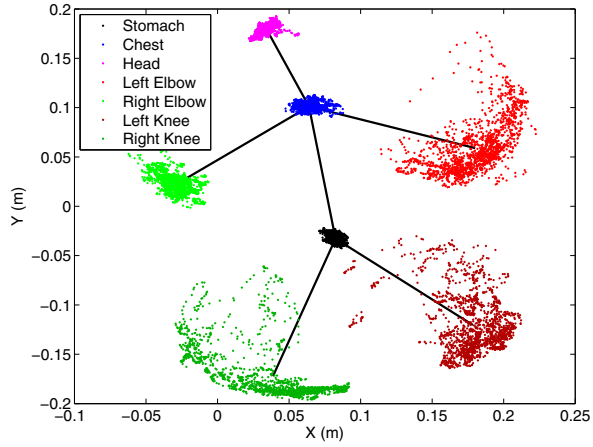


Fig. 8. Visualization of the location of 7 bodyjoints, namely; stomhach, chest, head, left/right elbows and left/right knee. The lower variation of the right elbow, shows that less movement has occurred for this joint.

4 Conclusion

We have described a method for 3D motion tracking of infants, based on a model-based approach. We show how this method gives good results, with respect to both accuracy and precision, compared to a novel study on motion tracking of infants. The method is based on fitting a 3D model to the observed 3D data, obtained with the Microsoft Kinect sensor. The model is combined by a number of geometric shapes that are connected based on a hierarchical skeleton model. We show how the results can be used to assess the motion pattern of infants by evaluating the raw motion parameters or the spatial 3D motion paths. This is a step closer to an automatic system that can help doctors assess the development of infants' motor control in order to detect motor impairing diseases such as cerebral palsy. In future work, we will focus on making the tracking more robust, such that the method is able to recover from difficult poses e.g. when the infants rolls over from supine to prone position.

Acknowledgments. The authors would like to thank the Helene Elsass Center and the Ludvig and Sara Elsass Foundation for funding the project as well as all of the infants and their families for participating in this project. Furthermore, the authors would like to thank the organization APA, for helping make contact with the participating families.

References

1. Himmelmann, K.: Epidemiology of cerebral palsy. Handbook of Clinical Neurology **111**, 163–167 (2013)
2. Bax, M., Goldstein, M., Rosenbaum, P., Leviton, A., Paneth, N., Dan, B., Jacobsson, B., Damiano, D.: Proposed definition and classification of cerebral palsy. Developmental Medicine & Child Neurology **47**(8), 571–576 (2005)
3. Goldsmith, S., Badawi, N., Blair, E., Taitz, D., Keogh, J., McIntyre, S.: A systematic review of risk factors for cerebral palsy in children born at term in developed countries. Developmental Medicine and Child Neurology **55**(6), 499–508 (2013)
4. McIntyre, S., Morgan, C., Walker, K., Novak, I.: Cerebral palsy-don't delay. Dev Disabil Res Rev **17**(2), 114–29 (2011)
5. Murphy, N., Such-Neibar, T.: Current problems in pediatric and adolescent health care. In: Cerebral Palsy Diagnosis and Management: The State of the Art, pp. 149–69 (2003)
6. Einspieler, C., Prechtl, H., Bos, A., Ferrari, F., Cioni, G.: Prechtl's Method on the Qualitative Assessment of General Movements in Preterm. Wiley, Term and Young Infants. Clinics in Developmental Medicine (2008)
7. Berg, A.: Modellbasert klassifisering av spedbarns bevegelser (2008)
8. Rahmanpour, P.: Features for movement based prediction of cerebral palsy (2009)
9. Meinecke, L., Breitbach-Faller, N., Bartz, C., Damen, R., Rau, G., Disselhorst-Klug, C.: Movement analysis in the early detection of newborns at risk for developing spasticity due to infantile cerebral palsy. Human Movement Science **25**(2), 125–144 (2006)

10. Karayiannis, N.B., Varughese, B., Tao Jr., G.: J.D.F., Wise, M.S., Mizrahi, E.M.: Quantifying motion in video recordings of neonatal seizures by regularized optical flow methods. IEEE Transactions on Image Processing **14**(7), 890–903 (2005)
11. Stahl, A., Schellewald, C., Stavdahl, O., Aamo, O.M., Adde, L., Kirkerod, H.: An optical flow-based method to predict infantile cerebral palsy. IEEE Transactions on Neural Systems and Rehabilitation Engineering **20**(4), 605–614 (2012)
12. Olsen, M.D., Herskind, A., Nielsen, J.B., Paulsen, R.R.: Motion tracking of infants. In: 22nd International Conference on Pattern Recognition (ICPR). (2014, to appear)
13. Plagemann, C., Ganapathi, V., Koller, D., Thrun, S.: Real-time identification and localization of body parts from depth images. In: 2010 IEEE International Conference on Robotics and Automation (ICRA), pp. 3108–3113 (2010)
14. Baak, A., Müller, M., Bharaj, G., Seidel, H.P., Theobalt, C.: A data-driven approach for real-time full body pose reconstruction from a depth camera. In: IEEE 13th International Conference on Computer Vision (ICCV), pp. 1092–1099. IEEE (November 2011)
15. Schwarz, L.A., Mkhitaryan, A., Mateus, D., Navab, N.: Estimating human 3d pose from time-of-flight images based on geodesic distances and optical flow. In: FG, pp. 700–706. IEEE (2011)
16. Ganapathi, V., Plagemann, C., Koller, D., Thrun, S.: Real-time human pose tracking from range data. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part VI. LNCS, vol. 7577, pp. 738–751. Springer, Heidelberg (2012)
17. Droeschel, D., Behnke, S.: 3d body pose estimation using an adaptive person model for articulated icp. In: Jeschke, S., Liu, H., Schilberg, D. (eds.) ICIRA 2011, Part II. LNCS, vol. 7102, pp. 157–167. Springer, Heidelberg (2011)
18. Eberly, D.: Distance from a point to an ellipse, an ellipsoid, or a hyperellipsoid (2013)
19. Portal, R., Dias, J., de Sousa, L.: Contact detection between convex superquadric surfaces. Archive of Mechanical Engineering LVII **2**, 165–186 (2010)
20. Mnier, C., Boyer, E., Raffin, B.: 3d skeleton-based body pose recovery. In: 3DPVT, pp. 389–396. IEEE Computer Society (2006)
21. Shen, S., Tong, M., Deng, H., Liu, Y., Wu, X., Wakabayashi, K., Koike, H.: Model based human motion tracking using probability evolutionary algorithm. Pattern Recognition Letters **29**(13), 1877–1886 (2008)
22. Moeslund, T.B., Hilton, A., Krger, V., Sigal, L., eds.: Visual Analysis of Humans - Looking at People. Springer (2011)
23. Pons-Moll, G., Rosenhahn, B.: Ball joints for marker-less human motion capture. In: IEEE Workshop on Applications of Computer Vision (WACV) (2009)

Appendix C

Using Motion Tracking to Detect Spontaneous Movements in Infants

Accepted for poster presentation at the Scandinavian Conference of Image Analysis 2015 in Copenhagen.

Using Motion Tracking to Detect Spontaneous Movements in Infants

Mikkel D. Olsen( ), Anna Herskind, Jens Bo Nielsen, and Rasmus R. Paulsen

Department of Applied Mathematics and Computer Science, Technical University of Denmark, Richard Petersens Plads, Building 324, 2800 Kgs. Lyngby, Denmark
mdol@dtu.dk

Abstract. We study the characteristics of infants' spontaneous movements, based on data obtained from a markerless motion tracking system. From the pose data, the set of features are generated from the raw joint-angles of the infants and different classifiers are trained and evaluated using annotated data. Furthermore, we look at the importance of different features and outline the most significant features for detecting spontaneous movements of infants. Using these findings for further analysis of infants' movements, this might be used to identify infants in risk of cerebral palsy.

Keywords: Motion analysis · Motion tracking · Movement classification · Motion features

1 Introduction

In the last decades, motion tracking has become more and more popular. Whether it is marker-based or markerless, vision-based or sensor-based, the common goal is to estimate the pose and movement of people. Since the introduction of the Microsoft Kinect depth sensor in 2010, motion tracking has become a relative easy problem to solve. Without much effort, the underlying pose and motion parameters can be obtained and the next step is thus to utilize these parameters. In relation to the initial purpose of the Kinect sensor, the extracted pose parameters was used as input to the Microsoft XBox console, to control the character within a computer game. However, the list of applications is far more comprehensive. In [1,2] the pose estimation is used to extract features such as speed and step length. Features like these can be used for recognizing people, based on their gait, as shown in [3]. Other studies do not focus on recognizing a specific person, but instead recognizing different actions, such as walking, running, boxing, jumping, etc. [4]. However, common for most studies is that they focus on recognizing movements, that are easy to differentiate from each other, such as walking/jumping/punching/etc. Recently, new studies and challenges consider the concept of looking at more similar actions, such as recognizing sign language gestures, where two gestures can seem very similar to

R.R. Paulsen and K.S. Pedersen (Eds.): SCIA 2015, LNCS 9127, pp. 410–417, 2015.
DOI: 10.1007/978-3-319-19665-7_34

the untrained observer [5]. In this study, we focus on movements of infants. It is known that infants in the age of 3-5 months have special movements called fidgety movements [6]. Among high-risk infants such as infants born preterm, absent or abnormal fidgity movements is a strong indicator for the motor disorder cerebral palsy. Doctors are thus able to identify a high risk of cerebral palsy, in the early months after birth, based on assessing these special movements. Small movements in the trunk, neck and limbs characterize these special movements. The movements are easiest to detect when the infant is lying on its back, unstimulated [7]. However, kicking and crying influences the infants' movements and the fidgety movements will be obscured by these larger movements. Moreover, a pacifier can completely dampen the strength of the fidgety movements. In order to be able to recognize the fidgety movements, one approach is to first detect and remove the sequences where the fidgety movements do not appear and secondly to classify the remaining movements. In this paper, we focus on the first step, where we classify sequences of motion data of awake infants, with the goal of segmenting the sequence into segments of spontaneous/non-spontaneous movements. The classification is based on features obtained from a vision based and markerless motion tracking approach. A number of previous studies focus on quantifying these spontaneous movements. In [8] the authors quantify spontaneous kicks by tethering the legs to a mobile stand. When the infant kicks, the mobile moves and this information is used for further analysis. In [9], a similar mobile system is combined with a 2D based motion tracking system. Using both the mobile-observations as well as the motion tracking results, the kicking frequency can be obtained. In this study, the goal is to;

1. Test different classifiers for segmenting spontaneous movements, based on data extracted from a markerless motion tracking system.
2. Examine the importance of different movement based features in order to classify spontaneous movements.

2 Methods

2.1 Motion Data

The data used in this study are temporal RGB-D data obtained with the Microsoft Kinect sensor. The recorded data contains both color and depth information of infants in the age of 1-6 months (corrected with respect to term). During the recording-session, the infant was positioned on a mat, while the RGB-D camera was positioned above the infant. No equipment was attached to the infant and the infant was thus able to move freely and unaffected. However, it was required that the infant only wore a short-sleeved bodystocking, in order to see the joints of the over- and under-extremities. Furthermore, the infant was in a good mood and unstimulated during the recording. The infant's parents were informed about the procedure beforehand and at any time; the parents could choose to stop the session. Unless the session was interrupted, the recording was done for minimum 5 minutes.

2.2 Motion Tracking

The pose estimation and motion tracking of the infants are obtained using a previously developed system [10,11]. To summarize, the system fits an articulated 3D model to the 3D data obtained from the depth sensor. The model is structured from a set of relative simple 3D structures, namely cylinders, spheres and superellipsoids. A set of parameters define the shape and orientation of these structures, which are length, radius and angle/direction with respect to their relative parent structure. The stomach/torso defines the root structure and all other structures are connected either directly or indirectly to this structure. The fitting process is done by adjusting the orientation parameters, while minimizing the error-metric between the 3D data and the 3D model. The error-metric is simply based on the Euclidean distance between the model and the data. Figure 1 illustrates an example for the resulting pose estimation.

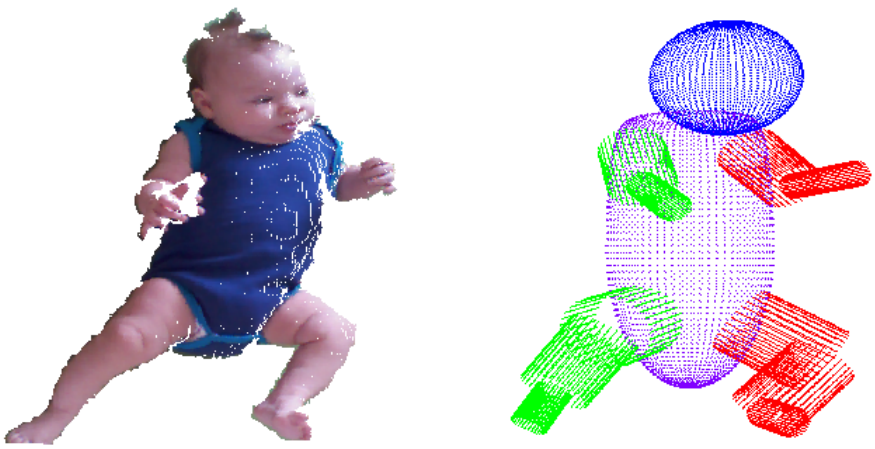


Fig. 1. Left: Colored point cloud obtained from the depth sensor. Right: 3D model fitted to the observed data.

2.3 Classification

The result from the motion tracking is a set of joint parameters, describing the pose of the recorded infant for each timestamp. The features used in this study are based on angular velocities and accelerations of the joints. For each frame, we calculate the angular velocities and accelerations, based on the joint angles in the current frame and the two previous frames. However, instead of using the raw data from a single frame, we transform the data using a sliding window approach, where we both generate mean- and median-filtered joint features. The transformations used for this approach are average, median, variance and the

Frobenius norm. Furthermore, we take the max/min values of the filtered velocities/accelerations, as we are interested in detecting frames where the infant is lying still vs. frames where the infant is doing an extreme movement with any part of the body. For classification methods we use *K-Nearest Neighbors (KNN)*, *Support Vector Machine (SVM)* and *Decision Tree*. For *KNN* we use two parameters for k, namely $k = 1$ and $k = 5$. In order to evaluate the different classifiers' performance, parts of the data have been annotated manually. The movements have been annotated either as spontaneous or calm.

3 Results

The dataset used in this study consists of 50k labelled frames taken from data recordings of 11 infants. For each frame, the frame was either labelled as being spontaneous or not. This labelling was done by one of the authors. Based on the four classification methods used in this study we examine how the training/test size influences the results. In Figure 2, the accuracy is shown, for different sizes of the training set. The test-set is simply the remaining data, when the training set has been extracted. For all four classification methods, the methods give good results, even with a small training set/large test set.

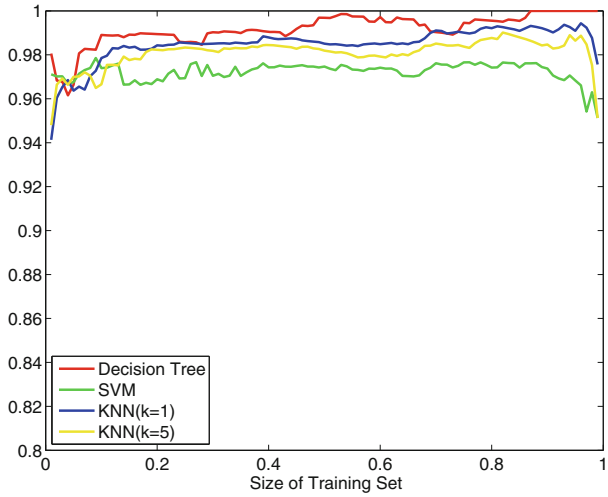


Fig. 2. The achieved accuracies for the four classifiers are illustrated, as a function of increasing size of the training set. The size is with respect to the total size of the data.

However, the classifiers are trained and tested without taking into account, that data trained from one infant is used to classify data from the same infant. We therefore train the classifiers on data from one infant, while the testing is done on data from the remaining infants. This is considered the worst case, as one could increase the size of the training set, by using more than one infant

for training. In Figure 3, the result can be seen, where cross-validation is used in order to consider training with all infants. Again, the size of the training set is varied, but in this case, the size of the total data set is related to the particular infant. This new choice of training/test sets yields an overall decrease in accuracy, as expected, but we are still able to obtain satisfactory results.

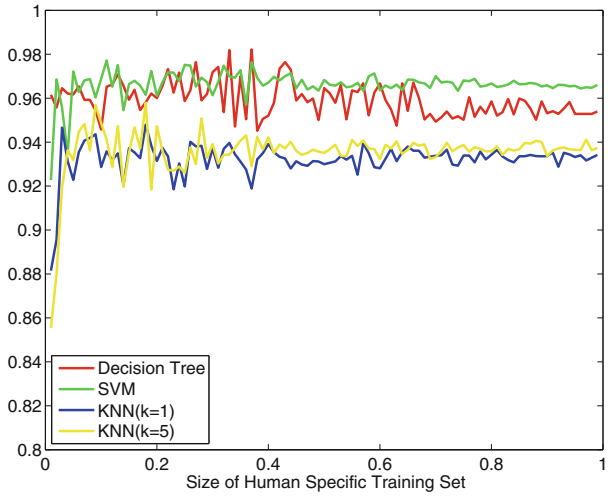


Fig. 3. For a more fair result, the training set is only based on data from one infant and the test set is based on data not belonging to the same infant. Cross-validation is used to train the classifiers on each infant.

Using a leave-one-out approach, we train a decision tree (the classifier that obtained the best results above) on data from all but one infant and segment the data from the leaved-out infant. The segmentation can be seen in Figure 4 where the ground truth segmentation can be compared with the estimated segmentation. Doing this for all infants, i.e. training the model on all but one infant and test the model on the leaved out infant, we are able to estimate the mean and standard deviation of the accuracy, which shows how good the methods are to generalize to an unknown infant. This has been done for all four methods and the results can be seen in Figure 5. We observe that we are still able to obtain good results for all four methods.

3.1 Parameter Importance

In order to point out the most important features used for detecting spontaneous movements, we use a leave-one-out approach. By removing one feature and training the classifiers, we compare the accuracy with the result obtained with the full set of features. This is done using 10-fold cross-validation. Figure 6 shows the results for the four types of classifiers. It should be noted that the importance-quantity has been normalized. It can be seen that the most important features

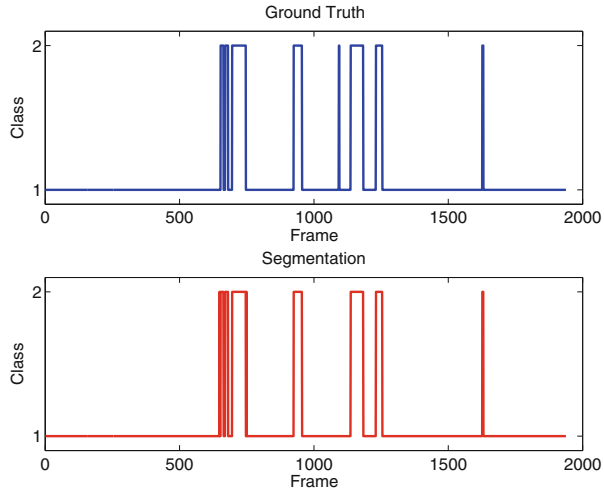


Fig. 4. Comparison between ground truth and estimated segmentation. Class 1 is spontaneous movements and Class 2 is non-spontaneous movements. Top: The ground truth segmentation of one infant's spontaneous movements. Bottom: The estimated segmentation of the infant's spontaneous movements.

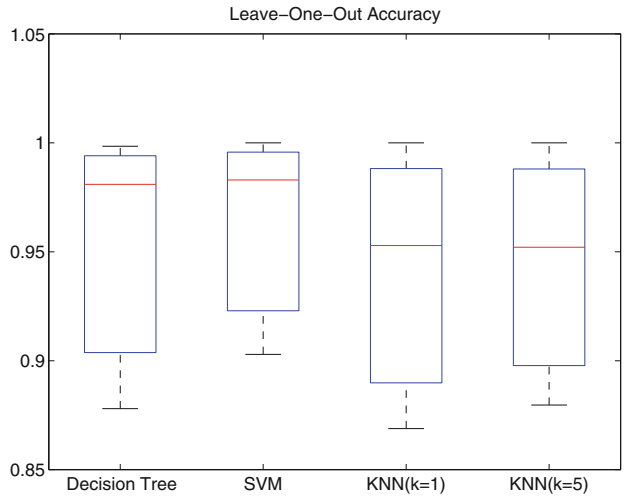


Fig. 5. Results for doing leave-one-infant-out experiment.

are based on velocities and that the maximum and summed velocity in a time windows significantly characterizes the spontaneous movements.

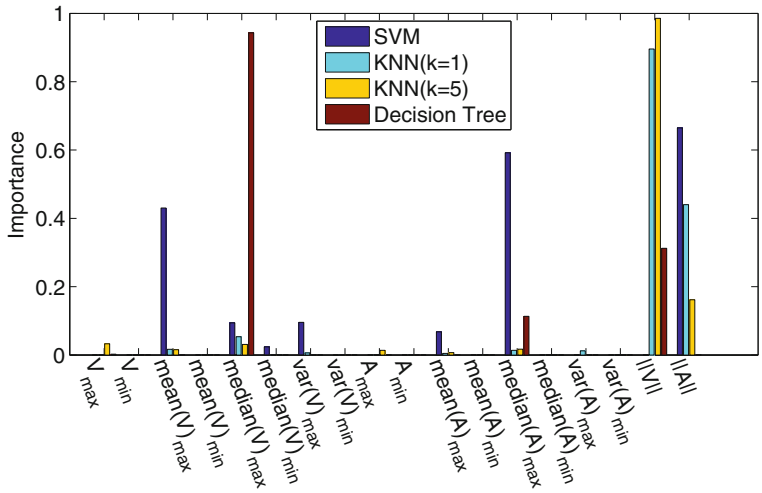


Fig. 6. The importance of the different parameters are illustrated for the four classifiers, based on the leave-one-out approach

4 Conclusion

Using annotated motion tracking data of moving infants in the age of 1-6 months, we have been able to segment sequences of spontaneous movements in infants. This was done using four different classifiers, which all proved to obtain similar results, ranging from 92−98% accuracy, based on different classifiers and different sizes of the training/test data. In addition, we evaluated the importance of the different features used in this study, where the maximum velocity and summed velocity over time are important features for the spontaneous movements.

Acknowledgments. The authors would like to thank the Helene Elsass Center and the Ludvig and Sara Elsass Foundation for funding the project as well as all of the infants and their families for participating in this study.

References

1. Jensen, R.R., Paulsen, R.R., Larsen, R.: Analysis of gait using a treadmill and a time-of-flight camera. In: Kolb, A., Koch, R. (eds.) Dyn3D 2009. LNCS, vol. 5742, pp. 154–166. Springer, Heidelberg (2009)
2. Gabel, M., Renshaw, E., Schuster, A., Gilad-Bachrach, R.: Full body gait analysis with kinect. In: Proceedings of EMBC 2012 (2012)
3. Little, J., Boyd, J.: Recognizing People by Their Gait: The Shape of Motion (1996)

4. Zhou, F., De la Torre, F., Hodgins, J.K.: Hierarchical Aligned Cluster Analysis for Temporal Clustering of Human Motion. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) (2013)
5. Agarwal, A., Thakur, M. K.: Sign language recognition using Microsoft Kinect (2013)
6. Einspieler, C., Prechtl, H.F.: Prechtl's assessment of general movements: a diagnostic tool for the functional assessment of the young nervous system. Mental Retardation and Developmental Disabilities Research Reviews (2005)
7. Einspieler, C., Prechtl, H.F.R., Bos, A.F., Ferrari, F., Cioni, G.: Prechtl's Method on the Qualitative Assessment of General Movements in Preterm. Term and Yong Infants (2005)
8. Heathcock, J.C., Bhat, A.N., Lobo, M.A., Galloway, J.C.: The relative kicking frequency of infants born full-term and preterm during learning and short-term and long-term memory periods of the mobile paradigm. Physical Therapy (2005)
9. Landgraf, J.F., Tudella, E.: Effects of external load on spontaneous kicking by one and two-month-old infants. Brazilian Journal of Physical Therapy (2008)
10. Olsen, M.D., Herskind, A., Nielsen, J.B., Paulsen, R.R.: Body-part tracking of infants. In: 22nd International Conference on Pattern Recognition (2014)
11. Olsen, M.D., Herskind, A., Nielsen, J.B., Paulsen, R.R.: Model-based motion tracking of infants. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014 Workshops. LNCS, vol. 8927, pp. 673–685. Springer, Heidelberg (2015)

Appendix D

Scene Flow on Human Motion Data

Submitted to International Journal of Computer Vision - Special Issue on Machine Vision Applications

IJCV Special Issue on Machine Vision Applications manuscript No.
(will be inserted by the editor)

Scene Flow on Human Motion Data

Gudmundur Einarsson · Mikkel D. Olsen · Line H. Clemmensen · Rasmus R. Paulsen

Received: date / Accepted: date

Abstract Availability of modern RGB-D sensors has recently spiked an interest in scene flow estimation methods. These methods allow for extraction of features from RGB-D sequences, which are relevant for action recognition and clinical research on human movement. The contributions of this work are threefold. We propose a novel method (SHuMo method) to estimate scene flow on data including the motion of a single individual from an RGB-D sensor. In order to assess the performance of our, we generate a synthetic human motion data set with ground truth scene flow. We compare the performance to recently published state of the art scene flow estimation methods. Speed and accuracy are of utmost importance, thus we discuss the pros and cons of each method and compare their performance with standard metrics.

Keywords Scene-Flow · Human Motion · Performance Analysis · Feature Extraction · RGB-D sensors

1 Introduction

Scene flow (SF), also known as range flow, is the estimation of motion fields in a sequence of images where information about the distance of the pixels from the sensor (a depth map) is also available. The term was first introduced by Vedula et al [30] where a multi-camera setup was used to estimate the depth map. Most of the research on SF extends earlier methods developed to estimate optical flow (OF) [16]-[19], which is the estimation of a motion field in the image plane. The OF methods work in absence of depth information. Acquiring more information about the scene from the depth map also yields better estimation of the OF, where SF can be projected onto the image plane to yield OF. A good summary of OF algorithms can be found in [2] and a more recent one in [1]. We provide a SF method specifically tailored for human motion data and we summarize recently published state of the art methods used for SF estimation. This summary gives an overview of which methods are most appropriate for estimating SF on human motion data. Human motion data is hard to generate with ground-truth information. We have constructed a synthetic data set by tracking the skeleton of a person using the skeleton tracking [28] developed for Microsoft Kinect. These skeleton coordinates were fitted to a 3D avatar and the ground-truth SF was therefore obtainable. This data set portrays a range of motions in all dimensions of 3D space, thus providing a new alternative benchmark on SF performance. The data set also includes real human motion sequences. The data set is called SHuMo (Synthetic Human Motion) and is publicly available [1]. We encourage researchers of SF methods to benchmark their methods on this data set.

A data set showing artificial human motion with ground-truth is desired to have in assessing the performance of SF methods. This is mainly because of the

G. Einarsson (✉)
E-mail: guei@dtu.dk

M. D. Olsen
E-mail: mdol@dtu.dk

L. H. Clemmensen
E-mail: lkhc@dtu.dk

R. R. Paulsen
E-mail: rapa@dtu.dk

DTU Compute, Technical University of Denmark, DK-2800 Lyngby

[1] link to where the data is stored

rise in research where modern RGB-D sensors are used to acquire motion data on people, e.g. w.r.t. monitoring recovery [31], postural control [6], rehabilitation [5], Cerebral Palsy [22], or Multiple Sclerosis [18] just to name a few. There are many ways to construct features from this kind of data. One option in particular is to track the skeleton of a person and limit the features to that data. This gives a good dimensionality reduction from the original data space and removes a lot of redundant information. It may also remove finer subtle movements which are of interest, for example tremors, small vibrations and other small changes that might have a significant effect on the results of the study at hand. SF gives an opportunity to incorporate features that represent these effects. Furthermore, if such changes occur in a small part of the scene, the SF does not need to be computed for the whole image. This has been done for OF in action recognition, e.g. [10].

1.1 Brief Overview of SF

Although the first method introducing SF estimation was presented in a paper in 1999, it was not until later that this corner of computer vision acquired more attention. Affordability of modern RGB-D sensors is probably the main reason for this interest. This is joined with the availability of better hardware to deal with the calculations that follow these methods. The most notable sensor is the Microsoft Kinect which was first announced in June 2009. The first paper dealing with the estimation of SF using the Kinect [13] was published in 2011, several papers followed and more papers are still being published.

For validation of the SF methods, the authors either generated their own data sets to display results, or they generated or used synthetic data sets which show some simple motions. In some cases they additionally reported results on the Middlebury data set [27] and compared to its ground-truth data, which is available and generated using structured light with very high precision. The Middlebury data set has become somewhat a golden standard to compare to, but it does not contain examples of human like motion.

To solve SF one needs the pixel intensities and the depth values, but that is not enough. If a restriction is posed to only use this information the problem cannot be solved effectively, it becomes under-determined or ill-posed. Some form of regularization is needed to address the local structure of the flow, i.e. points close in space are likely to move similarly. There are many different kinds of regularization approaches that can be used well summarized in [1]. This gives rise to a variety of ways to attack this problem. There are also other things that can be varied. The following list gives an overview of the main different parts of SF estimation that differ between the methods that are explored in this paper.

- Setup of the data, is it regarded as a point cloud or images and depth maps.
- Data term, the data in the data term can consist of intensities, gradients and other features. Various norms are also used.
- Regularization/prior term.
- Optimization.
- Miscellaneous, e.g. usage of learning or color information.

From this list, it can be seen that these SF methods use techniques from various other fields. This may look like a sort of a framework that categorizes SF, but there is potential for methods that do not follow this schema. The abundance of different approaches possible to estimate SF justifies the reason to study it, since different methods may not work the same on data from different scenes. In almost all cases there is also some parameter tuning needed e.g. the weights of regularization terms.

OF methods were initially characterized in [2]. The characterization is fundamentally w.r.t. how the methods are formulated, e.g. differential techniques, region-based matching and energy-based methods. The summary in [1] gives a finer break down of individual components of the methods and characterizes the methods based on how each component is constructed, e.g. what norm is used for the data-term, what kind of optimizations is used etc. Most of the reviewed methods in this paper are energy-based but more descriptions are given for the individual cases and highlighted in Table 1.

1.2 Structure of Paper and Contributions

The rest of the paper is organized as follows:

- Description of the SHuMo method.
- Overview of the methods (Section-3). This includes more detailed descriptions of the other methods we compare to and comments regarding potential for parallelization or availability of code.
- Description of the SHuMo data set (Section-4). This section contains information on how the data was generated and why the specific movements were selected to use in the data set.
- Description of experimental comparison (Section-5). This section summarizes the different error measurements used to compare the methods.
- Results (Section-6). This section includes comparisons from the results of the methods to the ground truth and figures showing how the methods deviate from the ground-truth.

– Discussion (Section-7), conclusion (Section-8) and an appendix with additional figures.

The main contributions of this article are a creation of a SF method for RGB-D data on human motion, a comparison of recently published state of the art SF methods on human motion data and the creation of a novel data set to compare SF methods.

2 The SHuMo method

The SHuMo method is not a general SF method. It is specifically designed to extract SF from human motion data, thus it only works on cases where a human is present in the scene with a static RGB-D sensor. The goal of the method is to extend the information/features obtainable from human motion tracking by calculating the flow of segmented body parts of the human.

The human motion tracking from [22] is used to generate a region of interest in the image domain. The motion tracking is based on an articulated human model and fits ellipsoids and cylinders to the different body parts using a hierarchical model, thus providing individual masks for the following:

Head
Torso
Upper arm
Lower arm and hand
Upper leg
Lower leg
Feet

An example of the fitted model can be seen in Figure 1.

For a consecutive pair of frames in a video sequence, this model is fitted. The fitting of the model takes 0.4-0.5 seconds. Each limb in the first frame is then normalized w.r.t. the orientation of the limb in the next frame. For a given limb we have a point cloud in the first and second frame from the depth maps. After the point cloud from the first frame has been transformed w.r.t. to orientation we find the nearest neighbour in the second point cloud. This correspondence of points defines the flow between frame pairs and by default, the out is the scene flow between the two frames, measured in meters.

The assumptions made for this method are that the sensor position is fixed, there is a human in the scene and the nearest neighbour in aligned point clouds represents the SF correspondence between the frame pairs. The runtime of the method is 0.3 seconds per frame pair on a standard workstation, so a total of 0.7-0.8 seconds with the skeleton model fitting. Both the skeleton fitting and the flow calculations are implemented to run on a GPU.

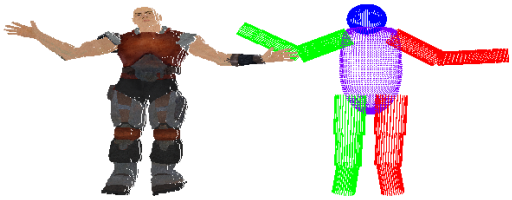


Fig. 1 Image showing the model fitted to the data (with an offset for better visualization). The model fits the data and is represented by a list of position-, size- and orientation-parameters.

This approach has some issues when occluded points come into view, as the neighbourhood of the points is not used to regularize the flow.

3 Overview of State of The Art Methods

In this paper nine methods, including ours (see Section 2), are evaluated on the SHuMo data set (See Section 4). These methods were selected due to availability of code or the fact that the authors of these methods were willing to try their methods on SHuMo. Some of these methods have been parallelized for calculations on a GPU which give run-time improvements. Small summaries of the methods are presented in Table 1. The selected methods give an overview of the potential of modern SF estimation methods.

3.1 Notation

Different authors tend to use different notation for the methods. This paper aims at unifying the notation to some extent. The following summarizes the main notation, although individual methods may deviate from this w.r.t. to their specific technicalities.

The displacement vector in the SF vector field is denoted $\mathbf{U} = (u, v, w)^T$ and the corresponding OF vector is denoted $\mathbf{u} = (u, v)^T$. Sometimes $\mathbf{U}(x, y)$ or $\mathbf{u}(x, y)$ is used to indicate flow at a specific point in the image domain. $\mathbf{U}$ is in the world coordinates and $\mathbf{u}$ is in pixels. The placement of a pixel is $\mathbf{x} = (x, y)^T$ and a scene point is represented as $\mathbf{X} = (x, y, z)^T$ in world coordinates, where z is the depth at position $(x, y)^T$ in the depth image Z. I and Z denote the RGB intensity image and the depth image respectively. The gradient

Table 1 Summaries from the methods. Two run times are shown for the first method, first is time when only depth images are used and second is when color images are also used. *The scene flow estimation was done by the author of the method and the run time cannot be compared directly with the other methods.

Name	Description	Availability of code	Run time/frame pair
SHuMo method (*our method*) (2015)	Region-based matching	Publicly available	0.3s
Computing Range Flow from Multi-modal *Kinect* Data (2011)	Differential global	Publicly available	15s & 100s
RGB-D Flow: Dense 3-D Motion Estimation Using Color and Depth (2013)	Energy-based global	Publicly available	30-80s
aTGV-SF: Dense Variational Scene Flow through Project Warping and Higher Order Regularization (2014)	Differential global	Not available	2.3s*
CP-Census: A Novel Model for Dense Variational Scene Flow from RGD-D Data (2014)	Patch-based global	Not available	2.95s*
Local/Global Scene Flow Estimation (2013)	Energy-based mixed	Author provided code	15s
Dense Semi-Rigid Scene Flow Estimation from RGBD images (2013)	Energy-based mixed	Author provided code	120s
A Primal-Dual Framework for Real-Time Dense RGB-D Scene Flow	Energy-based mixed	Publicly available	0.042s

operator ∇ applied to I or Z is in most cases the spatial gradient in the image domain. The derivatives w.r.t. time are denoted I_t and Z_t and a subscript of x and y denote the spatial derivatives. A subscript of 1 or 2 denotes the first and second frame in the images where the SF is estimated. The frame pairs of color and depth are denoted $\{I_1, Z_1\}$ and $\{I_2, Z_2\}$. Ω is the domain of the images and Ω^* is the part of the domain where the norm of the ground-truth SF is non-zero. Types of gradient filters are not always specified, but Sobel and Scharr filters [26] are used in some cases. $||\cdot||$ is the Euclidian norm and $|\cdot|$ is the absolute value of a scalar.

Another problem that comes up is the scale of the estimated flow in the depth direction. Authors tend to use either cm or m. There is not a general consensus on what is appropriate, but of course this is scene dependent, so a scale should be chosen w.r.t. the scene. When using a specific sensor like the Kinect, where the estimation of depth is limited to a certain range, one could decide on a specific scale to work on. This is not an issue for OF, since working with pixels makes it scale independent.

Yet another problem is that some methods output flow vectors $\mathbf{U}$ which have the first two components in pixels and the third in a metric scale. These methods output the OF with a depth flow component. The SHuMo data set has the ground-truth in cm for all components of the flow. Flow vectors that contain measurements in pixels for the first two components can be transformed to vectors where everything is in a metric scale if the camera matrix is known.

3.2 Computing Range Flow from Multi-modal *Kinect* Data (2011)

This paper [13] is the first one that addresses SF with data from the Kinect sensor. Their main contributions are a novel channel alignment algorithm to reduce the size of image areas which lack valid measurements and an extension of scene flow approaches to deal with unstable depth estimates. Note that range flow is another term used for scene flow.

The authors give a general description on how they expand the SF computations. They propose a method where in general any global optical flow algorithm can be used with an added data-term for the depth. They choose a refined version of the method originally proposed by Horn and Schunk [16] for their experiments. This method is thus a differential one. The depth data-term is formulated in the following equation:

$$\nabla Z \cdot \mathbf{u} + w + Z_t = 0. \tag{1}$$

Eq. 1 is known as the *range flow motion constraint* (RFMC).

The implementation is done with a standard image pyramid. The computations of SF for each level corresponds to minimizing the energy functional:

$$\begin{aligned} \int\int \big[&(I_x u + I_y v + I_t)^2 + \\ &\lambda_Z(x,y)(Z_x u + Z_y v + w + Z_t)^2 + \\ &\lambda_R(x,y)(|\nabla u|^2 + |\nabla v|^2 + |\nabla w|^2)\big] dxdy \end{aligned} \tag{2}$$

The deviations from OF is the factor $(Z_x u + Z_y v + w + Z_t)^2$ in the 2nd term (the RFMC), the term $|\nabla w|^2$ inside the third term (regularization for the range flow component) and the regularization weight functions λ_Z and λ_R in front of the 2nd and the 3rd terms. The

λ_Z function takes two values, 0 or a constant $c_Z > 0$ depending on whether depth measurements exist, i.e. the 2nd term inside the integral in Eq. 2 is omitted if no depth information is available in Z_1 or Z_2. The weight function λ_R takes the value c_{R_1} where λ_Z is positive, otherwise it takes the value c_{R_2} where $c_{R_1} >> c_{R_2}$. So stronger regularization is imposed on regions with depth data.

The authors state that weak regularization on invalid regions leads to sharper motion borders on edges. This may seem counter intuitive but having no depth information is a sign of depth discontinuities, therefore the scene flow is not over-smoothed around boundaries of moving objects.

The software is available online in the open source Charon framework[2]. There are two method available, one that uses only depth and another that uses both color and depth. Results from both methods are shown.

The authors mention that the algorithm was not implemented on a GPU, which would significantly improve the run-time. At the time they wrote the paper a single run with two consequent frames took approximately 30 seconds.

3.3 RGB-D Flow: Dense 3-D Motion Estimation Using Color and Depth (2013)

The method presented in [15] extends a method for OF [4] which uses robust norms, nonlinearized data constraints and variational solutions, this approach is an energy-based one. Another focus of the article is to perform rigid motion segmentation, where flow is used to segment multiple objects in an active vision scenario. The authors mention that their estimate of movement in the depth direction is in meters.

The approach is to use a variational method to optimize the global energy functional E:

$$\mathbf{U}^* = \arg\min_{\mathbf{U}} \int_{\mathbf{x}\in I} F(\mathbf{x}, \mathbf{U}) d\mathbf{x} \tag{3}$$

The functional F includes data terms for color and depth as well as smoothness terms. The data term for color is created from the *optical flow constraint equation*:

$$\Delta_t I \approx I_2 + \nabla I_2 \cdot \mathbf{u} - I_1 \tag{4}$$

The Charbonnier penalty ψ is applied, which is a convex approximation of the L_1 norm and inherits the property of dealing with outliers. The data term for the color is therefore $E_C(\mathbf{u}) = \psi((\Delta_t I)^2)$. The depth data term $E_Z(\mathbf{U})$ is created similarly with respect to the *range flow constraint equation*. A factor $\mu(\mathbf{x}) = \frac{\sigma_Z(1)}{\sigma_Z(d)}$ is added which weighs the depth data term against the color data term. This is done because of the uncertainty in the measured depth. $\sigma_Z(d)$.

The regularization imposed consists of two terms in the functional F, namely a flow magnitude penalty $E_B(\mathbf{U})$ and a smoothness term:

$$E_S(\mathbf{U}) = \psi(|\nabla u|^2 + |\nabla v|^2 + \beta(\mathbf{x})|\nabla w|^2) \tag{5}$$

Here $\beta(\mathbf{x})$ is f^2, where f is the focal length of the color camera and ψ is again the Charbonnier penalty. When optimizing the functional in Eq. 5 the gradient of the flow field only comes in via the Laplacian of the flow field. While the smoothing term in Eg. 5 is isotropic, the authors also experiment with anisotropic smoothing which produces less smoothing across object boundaries. F therefore consist of the two data terms and the two regularization terms, where the regularization terms are multiplied by scalar parameters α and γ to control their influence.

The authors present results where they compare the different kinds of regularization and another method that computes optical flow independently and then they estimate the depth flow from the displacement in the optical flow. The runtime of the presented method is reported to be 8-30 seconds for a frame pair at 320x240 resolution. The largest problem that the authors encountered was occlusion handling, i.e. pixels that disappear are assigned large flow vectors. The rest of the paper deals with object segmentation from motion.

3.4 aTGV-SF: Dense Variational Scene Flow through Project Warping and Higher Order Regularization (2014)

The method presented in [11] adds some new ideas on how to model SF, it is an energy-based method. The main contributions consist of (a) SF model with depth and intensity constraints where the temporal difference is calculated as a projection and back-projection in 3D space. (b) Regularization is coupled with anisotropic diffusion, (based on the input data), to better handle rotations and non-rigid movements. (c) The model is formulated as a convex energy minimization problem which can be parallelized efficiently.

To further capitalize on (a), the authors mention that they present the first method that estimates all components of the SF jointly in 3D space which improves handling of object magnification in image space. Due to this difference the formulation is described below.

[2] http://sourceforge.net/projects/charon-suite/

Fig. 2 Color images from the SHuMo data set. From left to right: first case (Forward), second case (Squat), third case (UpDown), fourth case (Walk).

The motion of each scene point $\mathbf{X}$ is given by $\frac{d\mathbf{X}}{dt} = \mathbf{U}$. The motion between consecutive scenes 1 and 2 is then defined as:

$$\mathbf{X}_2 = \mathbf{X}_1 + \mathbf{U} \tag{6}$$

The scene points are observed as projection at image position $\mathbf{x}$ in the image domain, with depth $D(\mathbf{x})$. So a scene point can be represented as $\mathbf{X} = K^{-1}\mathbf{x}^h D(\mathbf{x})$, where K is the camera matrix and h denotes homogeneous coordinates. The movement in image space between frames from time 1 and 2 can thus be formulated as in Eq. 7, which defines the warping W in image space:

$$\mathbf{x}_2 = W(\mathbf{x}_1, \mathbf{U})^h = \frac{K(K^{-1}\mathbf{x}_1^h D_1(\mathbf{x}_1) + \mathbf{U})}{D_1(\mathbf{x}_1) + w} \tag{7}$$

The authors use the temporal brightness consistency constraint in Eq. 8 with the projected and back projected point $\mathbf{x}_2$.

$$I_{\Delta t}(\mathbf{x}_t, \mathbf{U}) = I_{t+1}(W(\mathbf{x}_t, \mathbf{U})) - I_t(\mathbf{x}_t). \tag{8}$$

The authors use a linear approximation of the brightness difference at an initial flow field $\mathbf{U}_0$ applying a Taylor expansion which gives the intensity constraint in Eq. 9.

$$\begin{aligned} \rho_I(\mathbf{x}_1, \mathbf{U}, c) &= I_{\Delta t}(\mathbf{x}_1, \mathbf{U}_0) + \\ \nabla I_2(W_0)\frac{\partial W_0}{\partial \mathbf{U}_0}(\mathbf{U} - \mathbf{U}_0) &+ \delta c(\mathbf{x}_1) \end{aligned} \tag{9}$$

The abbreviations in Eq. 9 are W_0 for $W(\mathbf{x}_1, \mathbf{U}_0)$ and $I_{\Delta t}(\mathbf{x}_1, \mathbf{U}_0)$ is the brightness difference from Eq. 8 evaluated at $\mathbf{U}_0$. The variable $c(\mathbf{x})$ is incorporated into the model to compensate the violation of the brightness consistency introduced in [7], the parameter δ is introduced to control its effect. A depth constraint is also developed in a similar fashion.

As for the other methods, these two constraints are not enough to solve for the three variables in the SF. Instead of using the L_1 or the L_2 norm for the regularization the authors use a higher order model called total generalized variation (TGV) allowing for a reconstruction of piecewise polynomial functions. The TGV is defined in Eq. 10

$$TGV_\alpha^2 = \min_{\mathbf{U},\mathbf{v}} \alpha_1 \int_\Omega |\nabla \mathbf{U} - \mathbf{v}| dx + \alpha_0 \int_\Omega |\nabla \mathbf{v}| dx, \tag{10}$$

where $\mathbf{v}$ is introduced to impose second order smoothness. α_0, α_1 are weighing parameters. To improve the resulting flow field along strong depth borders the regularization term is additionally weighed with gradient information from the depth image D_1. This is done with an anisotropic diffusion tensor $T^{\frac{1}{2}}$ known as the Nagel-Enkelman operator [21]. The final energy for the optimization is:

$$\begin{aligned} \min_{\mathbf{U},\mathbf{v},c} \{ \int_\Omega & w(\lambda_I |\rho_I| + \lambda_D |\rho_D|) + |\nabla c|_\epsilon + \\ & \alpha_1 |T^{\frac{1}{2}}(\nabla \mathbf{U} - \mathbf{v})| + \alpha_0 |\nabla \mathbf{v}| dx \} \end{aligned} \tag{11}$$

The variable $w \in [0, 1]$ in Eq. 11 is a pixelwise confidence score from the depth sensor. The score is 0 where no depth information is available. The illumination c is also expected to be smooth and is therefore regularized with the parameterized Huber norm.

There is a total of eight parameters that can be adjusted and the authors report that the illumination parameters used for c and the parameters for TGV. Some more details on the formulation of the optimization can be found in the original paper.

The authors validate the algorithm on synthetic and real data and also the Middlebury data set. They also demonstrate very low runtime compared to other authors because of their GPU implementation. The authors ran the code on the SHuMo data set with the same parameters as for the Middlebury data set.

Currently the code is not publicly available.

3.5 CP-Census: A Novel Model for Dense Variational Scene Flow from RGB-D Data (2014)

Four of the authors from the last method also published another method [12] in 2014. The main differences lie

in the formulation of the data terms. To model the intensity data term the authors use the Ternery Census Transform (TCT) [32], this method had previously been applied to OF in [29]. This method is invariant to illumination changes and more robust to noise. This approach consists of calculating the TCT of both intensity images and then comparing them with the pixel-wise Hamming distance.

The depth term is calculated by directly matching 3D points. It is a flow error metric based on the Iterative Closest Point algorithm [33]. A patchwise point difference is calculated directly in 3D space to match local surface structure. When matching patches instead of depth pixel values the method becomes more robust for homogeneous depth regions and acquisition noise.

These data terms are highly non-convex in $\mathbf{U}$. A second order Taylor expansion is performed on the data terms around an initial flow field $\mathbf{U}_0$. A semi definite approximation of the Hessian is used where mixed derivatives are neglected, i.e. the Hessian is a diagonal matrix. The same regularization is proposed as for the last method above 3.4.

Other aspects of the method are similar to the one above 3.4 and the authors ran the code on the SHuMo data set with the same parameters as for the Middlebury data set.

Currently the code is not publicly available.

3.6 Local/Global Scene Flow Estimation (2013)

In the work by Quiroga et. al. [24], the authors combine a local approach [25], originally inspired by the Lucas-Kanade approach, with a global approach based on adaptive TV regularization. This is done by optimization of the cost function:

$$E(\mathbf{U}) = E_D(\mathbf{U}) + \alpha E_M(\mathbf{U}) + \beta E_R(\mathbf{U}), \tag{12}$$

where

- $E_D(\mathbf{U})$ is the data term that incorporates consistency between the observed intensity and depth information. This term is calculated using:

$$\begin{aligned}\mathbf{U} = \arg\min_{\mathbf{U}} \sum_{\mathbf{x}\in\Omega(\mathbf{x}_0)} \psi\left(\left[I(\mathbf{W}(\mathbf{x};\mathbf{U})) - T(\mathbf{x})\right]^2\right) \\ +\lambda\psi\left(\left[Z(\mathbf{W}(\mathbf{x};\mathbf{U})) - \left(T_z(\mathbf{x}) + D^T\mathbf{U}\right)\right]^2\right)\end{aligned} \tag{13}$$

- $E_M(\mathbf{U})$ defines a sparse matching consistency for a sparse set of 2D SURF feature points. However, instead of only using the matched feature locations, each pixel in the dataset is assigned to a pair of matched features points if the distance between the feature location and the pixel location is below some predefined threshold.
- $E_R(\mathbf{U})$ regularizes the scene flow using a TV based approach that preserves discontinuities and favors local rigidity. This term is defined as:

$$E_R(\mathbf{U}) = \sum_{\mathbf{x}} \omega(\mathbf{x}) |\nabla\mathbf{U}(x)|, \tag{14}$$

where $\omega(\mathbf{x})$ preserves strong discontinuities in the depth space and $|\nabla\mathbf{U}(x)|$ ensures locally rigid motions.

3.7 Dense Semi-Rigid Scene Flow Estimation from RGBD images (2014)

In [24] the authors described the scene flow as a field of 3D motion vectors, where the solution was a translation vector for each pixel in the scene. In [23], the authors extend their previous work, by describing the scene flow as a field of so-called twists, yielding both a translation and rotational term for each pixel in the data set.

Similar to their previous work, the authors optimize the cost function in Eq. 15, consisting of a data term and a regularization term.

$$E(\xi) = E_D(\xi) + \alpha E_S(\xi) \tag{15}$$

The equation is similar to Eq 12, except that the sparse matching is omitted and the result contains both translational and rotational information of the scene. Moreover, the data term $E(\xi)$ includes a gradient consistency term that preserves intensity changes. $E_S(\xi)$ is similar to $E_R(\mathbf{v})$ in Eq. 12, but the term is split into a translational part and a rotational part.

3.8 A Primal-Dual Framework for Real-Time Dense RGB-D Scene Flow (2015)

The article [17] presents a SF method that works very fast compared to other state of the art methods. The run-time reported in the paper is 0.042 second per frame pair, that is very close to real-time performance. These results are achieved using a highly parallelized implementation of a primal-dual algorithm in a variational framework. Regularization is imposed on the 3D surface, emphasizing that points in 3D space should move similarly.

The problem consists of minimizing a functional consisting of a data-term and a regularization term. The data-term:

$$E_D(\mathbf{U}) = \int_\Omega |\varrho_I(\mathbf{U},x,y)| + \mu(x,y)|\varrho_Z(\mathbf{U},x,y)|dxdy$$

Fig. 3 These images show the data for the first part of the data set (Forward), and the ground-truth. The images in the first column are the color data and the images in the second column are the depth data. The top image in the third column is the color coded ground truth OF and the bottom image is the flow in the depth direction, where blue represents movement away from the viewer and vice versa for red.

(16)

where ϱ_I enforces brightness consistency according to the optical flow constraint equation and ϱ_Z ensures geometric consistency according to the range flow constraint equation. $\mu(x, y)$ is a weighing factor on the geometric consistency term. The authors use the L_1 norm in the data-term.

The regularization term is similar to standard TV, but also takes into account the geometry of the scene. This enforces smooth similarity of nearby 3D points.

The authors compare their results to ones obtained from the RGB-D method [15]. They report better average performance and substantial improvements in runtime. The algorithm runs at 24 Hz, but there is an option to make the solver quit prematurely, yielding run-time performance of 30 Hz.

4 Description of Data

The following sections give an overview of how the SHuMo data set was created and what aspects of human motions are being captured within the different parts of the data set.

4.1 Creation of Synthetic Human Motion Data (SHuMo)

The first part in section 4.1.1 describes the creation of the part with the avatar and how the ground truth values were obtained. Section 4.1.2 describes the part with the real data and the creation of semi ground-truth values to compare to.

4.1.1 Avatar data

The avatar part of the data set contains depth and color images of various human motions (explained in Section 4.2). The first frame for the examples used in this paper can be seen in Figure 2. A more detailed version of the first case used can be seen in Figure 3, where both the depth and color frames can be observed, as well as the ground truth OF and depth flow. The OF is colored using the Middlebury color code, where color relates to direction and intensity relates to the magnitude of the flow-vectors.

In most studies, data are created by recording the depth and color images directly from the Kinect. However, this approach lacks in having ground truth information of SF in the scene. Alternatively, a 3D world can be simulated, e.g. by rendering a rotating box [11],[**?**] or ball[14], thus yielding full knowledge of the motion in the scene. However, these studies usually consider simple rotations and the complexity is limited. Inspired

Table 2 Data obtained for each frame in the synthesized data

Name	Description	type	Size
I	Color image	byte	$H \times W \times 3$
D	Depth image	short	$H \times W$
OFx	OF in X-direction (pixels)	float	$H \times W$
OFy	OF in Y-direction (pixels)	float	$H \times W$
SFx	SF in X-direction (cm)	float	$H \times W$
SFy	SF in Y-direction (cm)	float	$H \times W$
SFz	SF in Z-direction (cm)	float	$H \times W$

by these studies, we synthesize depth and color images of a moving 3D avatar, thus yielding complex human-like motions, while still being able to obtain the ground truth scene flow. The avatar is rendered in XNA using the *Avateering-XNA* [3] demo that comes with the Microsoft Kinect SDK for Windows, where the Kinect skeleton tracker [28] is used to control the avatar. For each detected skeleton pose, the avatar is transformed according to a number of bone-transformations. This yields a transformed 3D point cloud which can be projected on to a virtual 2D image plane. A projection matrix is used to project the 3D points into the 2D plane, based on the desired resolution of the output images, where W and H are the width and height (in pixels) respectively:

$$\mathbf{P} = \begin{pmatrix} W & 0 & W/2 \\ 0 & H & H/2 \\ 0 & 0 & 1 \end{pmatrix} \tag{17}$$

Knowing the location of every point in the previous frame enables tracking of the points and calculations of the flow in the 3D world and the 2D image plane. For each time instance, t, the data described in Table 2 are obtained. The OF describes the translation of each pixel in the image plane, while the SF is translation of points in the 3D world space. The flows are calculated with respect to the previous frame at time $t-1$.

As the avatar only fills out part of the color and depth image, the background is undefined. In order to obtain more realistic images, these missing regions are filled with a fixed background image. The background data is generated from a recording done in one of our office. Inpainting is used [8] in order to fill out missing holes in the fixed depth image. As the background is fixed, both the optical flow and the scene flow should be zero in these regions.

[3] https://msdn.microsoft.com/en-us/library/jj131041.aspx, SDK downloaded March 9th, 2015

4.1.2 Real data

The data set with real human motion consists of two movements, a forward walk and up and down swing of the hands. The color images were aligned to the depth images, this removes some of the color information from the color images, where depth information is not available. No ground truth data is available but we compare the results of other methods to the SHuMo method. This is not a true metric of an individual method's performance, but it is an indicator of how similar these methods are to the SHuMo method.

4.2 Overview of the different cases

The avatar data set consists of 4 types of movements. Each movement type has two consecutive color and depth images. We also release the full sequence behind every movement, it consists of 9 recordings with a duration of 2-4 seconds each yielding 60-120 frames for each case. These are not analysed in this paper, others can use them to try methods with temporal regularization or learning methods.

The real human data set consists of two types of movements. Again each movement type has two consecutive color and depth images. We also release a full sequence of movements with pseudo ground truth which spans 20 seconds of various movements. The choice for the different movements are based on the fact that many current studies on human action recognition from depth images, focus on action related to tracking the hands (sign language recognition) or movements involving the full body such as walking and jumping [20,9]. Furthermore, a set of complex movements was desired, involving the full body and some simpler movements, involving rotations and translations of body parts. The cases are divided into the following four groups:

- Forward(1-2): Simultaneously move right hand forward/backwards and the left hand in the opposite direction with a slight rotation of the torso.
- Squat(1-4): Lift hands while bending the knees.
- Updown(1-2): Simultaneously move right hand up/down and left hand in the opposite direction. Also includes real human data.
- Walk: Walk towards the left side of the camera with swinging arms. Also includes real human data.

Two successive color-depth pairs were selected from each of these four groups to test the methods in this paper (see Figure 2). A more detailed view of the first case can be seen in Figure 3. Similar figures for the other three cases are in the appendix. All the data are

Fig. 4 Image showing how the FDR works for results from a run of *Local/Global* on part 4 (Walk) of the data set. The white pixels indicate falsely discovered movement, i.e. effects of regularization. The proportion of white pixels is the FDR. Formula is shown in Table 3. The method was chosen randomly to illustrate the FDR.

available online with the ground truth [4]. The cases used in this paper are in a special directory.

There are two main reasons for selecting these specific movements, one is to capture motion in all dimensions of the 3D space instead of linear transformations of objects, another is to capture typical human motion such as the walk and the squat. The parts of the data set that regard the movement of the hands are also interesting. The methods may perform differently when only parts of the body are moving, this might lead to less uniform regularization.

5 Experimental Comparisons

The eight methods were tested on the four cases from the SHuMo data set. Three error quantities were calculated for SF. Two standard ones presented in [3] and false discovery rate (FDR). Some methods estimated movement in the static background, while others oversmoothed the movement around the boundaries due to regularization. The false discovery rate (FDR) summarizes how well the methods capture the movement in the region where there is actual movement, this is illustrated in Figure 4. The formula used for FDR is presented in Table 3.

Errors for OF are also reported. The errors are summarized in Table 3. The normalization used for $\mathrm{NRMS}_{\mathrm{SF}}$ is the magnitude of the largest ground truth SF vector. This was chosen due to the fact that it is easy to grasp the idea of how that scale relates to the magnitude of the error. This would not be applicable in scenes with very little motion. Note that the absolute angular error is assumed to be zero if either the ground-truth or the flow vector from a method is the zero vector, because the angle is not normally defined in that case. This creates some bias for the angular errors, due to all the zeroes, but the errors are still comparable between methods.

There is not a standardized way to estimate SF like in OF due to the different scales that the SF is estimated on, i.e. in world coordinates. The output from different methods varies, some give the SF in mm, others in cm or even m. The ground truth data in the SHuMo data set is given in cm. There are also different ways to normalize the errors.

6 Results

The main results are presented in Table 4. The best performing method gets a bold score and the second best score is underlined. There are a few things to note from the numerical results. The *RGB-D Flow* method consistently gets the best results on the OF error measurements. The *Local/Global* method is performing best in most cases on SF measures, but *Dense Semi-Rigid* and *CP-Census* are also giving similar results. *Primal-Dual Flow* and *SHuMo-Flow* are getting the best scores w.r.t. FDR. This is no surprise for *SHuMo Flow*, since it is estimating the region of movement before estimating the flow, and there is no regularization. The *Primal-Dual Flow* is doing very well in that regard and achieves on average pretty good results. It is hard to say that one method is strictly better than the other for this data, since the error measurements are quantifying different properties of the flow.

The *Range Flow* methods never acquire the best or second best score. The *aTGV-SF* method consistently estimates movement in the region where there is no ground-truth movement. That effect should only show up in the NRMS quantities for SF and OF, or possibly the FDR. Still it performs well in the angular measurements. The *Range Flow* methods perform rather poorly in general. The version of the method that uses both color and depth is better in most cases.

The FDR is in some cases higher than 50%. This is because the region where the ground-truth flow is nonzero sometimes contains flow vectors from the particular method which are smaller than the vectors which are estimated in the region where the ground-truth is zero. The only methods that obtain good FDR scores consistently are the ones that address the regularization

[4] link to data

Table 3 Error estimates used for evaluation of SF. N is the number of pixels in the image, Ω is the image domain and Ω^* is the part of the image domain where there is ground-truth motion. Zero subscript denotes the ground truth flow, $\mathbf{U}$ denotes SF in cm and $\mathbf{u}$ denotes OF. χ is the indicator function.

Name	Type of Flow	Quantity	Formula
NRMS_{SF}	SF	Percentage	$\frac{\sqrt{\frac{1}{N}\sum_{\Omega}\|\|\mathbf{U}(x,y)-\mathbf{U}_0(x,y)\|\|^2}}{\max_{(x,y\in\Omega)}\|\|\mathbf{U}_0(x,y)\|\|}$
AAE_{SF}	SF	Degrees	$\frac{1}{N}\sum_{\Omega}\left\|\arccos\left(\frac{\mathbf{U}(x,y)\cdot\mathbf{U}_0(x,y)}{\|\|\mathbf{U}(x,y)\|\|\cdot\|\|\mathbf{U}_0(x,y)\|\|}\right)\right\|$
FDR	SF	Percentage	$\frac{1}{N}\sum_{\Omega}\chi(\mathbf{U}_0(x,y)=\mathbf{0})\cdot\chi(\|\|\mathbf{U}(x,y)\|\|>\min_{(x,y\in\Omega^*)}\|\|\mathbf{U}(x,y)\|\|)$
NRMS_{OF}	OF	Pixel	$\sqrt{\frac{1}{N}\sum_{\Omega}\|\|\mathbf{u}(x,y)-\mathbf{u}_0(x,y)\|\|^2}$
AAE_{OF}	OF	Degrees	$\frac{1}{N}\sum_{\Omega}\left\|\arccos\left(\frac{\mathbf{u}(x,y)\cdot\mathbf{u}_0(x,y)}{\|\|\mathbf{u}(x,y)\|\|\cdot\|\|\mathbf{u}_0(x,y)\|\|}\right)\right\|$

such that it does affect the whole domain or use information regarding region of movement, such as *SHuMo Flow*.

Note, that no additional parameter tuning was performed on the methods. The methods were all ran out of the box. The parameters for *aTGV-SF* and *CP-Census* are the same as used in the corresponding papers for the Middlebury data set. This could probably improve the performance of some of the methods, but would require significant run-time in cases where there are many parameters that need to be tuned.

Additional figures (See Figures 8, 9, 10 and 11) are provided in the appendix, which demonstrate the difference between the estimated flow from the methods and the ground truth. These figures demonstrate that the fourth part of the data set (Walk) was the hardest. The whole body of the avatar is moving in one general direction. The total movement is captured by all the methods, but the finer details are not as well captured. This might be due to regularization. The methods perform better on the cases where there are isolated movements in certain regions of the avatar, like the moving of the hands. The squat is also a movement of the whole body, but the methods deal better with that. This may be because the squat movement is rather limited in the depth dimension.

7 Discussion

While we believe that the SHuMo data set is a valuable contribution to evaluating SF methods, there certainly are other possibly ways to create such a data set. It is possible to track the movement of a person in different ways and there is potential for using other types of avatars that might have different kinds of constraints for the movement. For instance, when the squat for the SHuMo data set was recorded the person performing the squat went much lower than the avatar. This is due to the restrictions imposed by the constraints in the model behind the avatar. It is possible to alleviate some of these factors. It would still not likely change the relative comparison of the different SF methods.

Another possibility was also to create the data set in higher resolution. A decision was made to create it in the same resolution as the output from the Kinect sensor, since the methods are developed to work on data from the Kinect.

One thing that can be noticed from the results is that newer methods seem to be performing better. This shows that there is active improvement in the development of SF methods.

The FDR error measurements does not deliver it's purpose for methods that consistently estimate flow vectors as non-zero in the still part of the scene. There it shows that the methods estimate flow in regions where there is no movement. It depends on the application whether that matters or not. It is of course possible to threshold the flow, but that will also remove some movement from regions where there is actual movement. Another options exist to filter out the region where there is actual movement, but fitting a human-model like in *SHuMo Flow* to create a mask for the region of interest works well. The run-time for fitting such a model is not significant compared to the run-time of most of the methods except for the *Primal-Dual Flow*, but that method is already doing very well in that regard.

The comparison was done over the whole image domain. The comparison could have been restricted to the domain where there is non-zero ground truth movement, i.e. a non-still part of the scene. This was not considered an option, since it is of interest to see ex-

actly how the methods perform in still regions as well. There may not be still regions in real footage because of acquisition noise but seeing how the methods perform on the still region of the scene gives possibilities of estimating a lower bound on the error one might expect from the methods performing on regions that are supposedly still.

The run times of the methods can be seen in Table 1. There are tremendous differences, but it is also a matter of whether the methods are implemented in parallel or not. If no temporal regularization is in the methods, then it is possible to split the process of one sequence on a cluster in a simple way.

8 Conclusions

The results from this paper show that there is active improvement in the development of SF methods. New sensors that produce better quality data will surely push further the development of SF methods. The methods evaluated show potential for estimating SF on human motion data. The results also show that if SF is used for human motion data, then information regarding that can be added to give better results. A data set that give other authors the possibility to see how their methods perform on human motion data is now publicly available. It is the our hope that future developers of SF methods benchmark their methods on the SHuMo data set.

9 Acknowledgments

We would like to thank the authors of the used methods/papers for providing access to their code or taking the time to apply their methods on our data.

References

1. Baker, S., Scharstein, D., Lewis, J., Roth, S., Black, M.J., Szeliski, R.: A database and evaluation methodology for optical flow. International Journal of Computer Vision **92**(1), 1–31 (2011)
2. Barron, J.L., Fleet, D.J., Beauchemin, S.S.: Performance of optical flow techniques. International journal of computer vision **12**(1), 43–77 (1994)
3. Basha, T., Moses, Y., Kiryati, N.: Multi-view scene flow estimation: A view centered variational approach. International journal of computer vision **101**(1), 6–21 (2013)
4. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: Computer Vision-ECCV 2004, pp. 25–36. Springer (2004)
5. Chang, Y.J., Chen, S.F., Huang, J.D.: A kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities. Research in developmental disabilities **32**(6), 2566–2570 (2011)
6. Clark, R.A., Pua, Y.H., Fortin, K., Ritchie, C., Webster, K.E., Denehy, L., Bryant, A.L.: Validity of the microsoft kinect for assessment of postural control. Gait & posture **36**(3), 372–377 (2012)
7. Cornelius, N., Kanade, T.: Adapting optical-flow to measure object motion in reflectance and x-ray image sequences. ACM SIGGRAPH Computer Graphics **18**(1), 24–25 (1984)
8. D'Errico, J.: inpaint_nans. MATLAB Central File Exchange, http://www.mathworks.com/ matlabcentral/fileexchange/ 4551-inpaint-nans (Retrieved March 9, 2015)
9. Escalera, S., Baró, X., Gonzalez, J., Bautista, M.A., Madadi, M., Reyes, M., Ponce-López, V., Escalante, H.J., Shotton, J., Guyon, I.: Chalearn looking at people challenge 2014: Dataset and results. In: Computer Vision-ECCV 2014 Workshops, pp. 459–473. Springer (2014)
10. Fanello, S.R., Gori, I., Metta, G., Odone, F.: Keep it simple and sparse: Real-time action recognition. The Journal of Machine Learning Research **14**(1), 2617–2640 (2013)
11. Ferstl, D., Reinbacher, C., Riegler, G., Ruther, M., Bischof, H.: atgv-sf: Dense variational scene flow through projective warping and higher order regularization. In: 3D Vision (3DV), 2014 2nd International Conference on, vol. 1, pp. 285–292. IEEE (2014)
12. Ferstl, D., Riegler, G., Rther, M., Bischof, H.: Cp-census: A novel model for dense variational scene flow from rgb-d data. In: Proc. of the British Machine Vision Conf.(BMVC), vol. 2 (2014)
13. Gottfried, J.M., Fehr, J., Garbe, C.S.: Computing range flow from multi-modal kinect data. In: Advances in Visual Computing, pp. 758–767. Springer (2011)
14. Hadfield, S., Bowden, R.: Scene particles: Unregularized particle-based scene flow estimation. Pattern Analysis and Machine Intelligence, IEEE Transactions on **36**(3), 564–576 (2014)
15. Herbst, E., Ren, X., Fox, D.: Rgb-d flow: Dense 3-d motion estimation using color and depth. In: Robotics and Automation (ICRA), 2013 IEEE International Conference on, pp. 2276–2282. IEEE (2013)
16. Horn, B.K., Schunck, B.G.: Determining optical flow. In: 1981 Technical symposium east, pp. 319–331. International Society for Optics and Photonics (1981)
17. Jaimez, M., Souiai, M., Gonzalez-Jimenez, J., Cremers, D.: A primal-dual framework for real-time dense rgb-d scene flow
18. Kontschieder, P., Dorn, J.F., Morrison, C., Corish, R., Zikic, D., Sellen, A., D?Souza, M., Kamm, C.P., Burggraaff, J., Tewarie, P., et al.: Quantifying progression of multiple sclerosis via classification of depth videos. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2014, pp. 429–437. Springer (2014)
19. Lucas, B.D., Kanade, T., et al.: An iterative image registration technique with an application to stereo vision. In: IJCAI, vol. 81, pp. 674–679 (1981)
20. Megavannan, V., Agarwal, B., Babu, R.V.: Human action recognition using depth maps. In: Signal Processing and Communications (SPCOM), 2012 International Conference on, pp. 1–5. IEEE (2012)
21. Nagel, H.H., Enkelmann, W.: An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. Pattern Analysis and Machine Intelligence, IEEE Transactions on (5), 565–593 (1986)
22. Olsen, M.D., Herskind, A., Nielsen, J.B., Paulsen, R.R.: Model-based motion tracking of infants. In: Computer

Vision-ECCV 2014 Workshops, pp. 673–685. Springer (2014)
23. Quiroga, J., Brox, T., Devernay, F., Crowley, J.: Dense semi-rigid scene flow estimation from rgbd images. In: Computer Vision–ECCV 2014, pp. 567–582. Springer (2014)
24. Quiroga, J., Devernay, F., Crowley, J.: Local/global scene flow estimation. In: Image Processing (ICIP), 2013 20th IEEE International Conference on, pp. 3850–3854. IEEE (2013)
25. Quiroga, J., Devernay, F., Crowley, J.: Local scene flow by tracking in intensity and depth. Journal of Visual Communication and Image Representation **25**(1), 98–107 (2014)
26. Scharr, H.: Optimal filters for extended optical flow. Springer (2007)
27. Scharstein, D., Szeliski, R.: High-accuracy stereo depth maps using structured light. In: Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on, vol. 1, pp. I–195. IEEE (2003)
28. Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., Cook, M., Moore, R.: Real-time human pose recognition in parts from single depth images. Communications of the ACM **56**(1), 116–124 (2013)
29. Stein, F.: Efficient computation of optical flow using the census transform. In: Pattern Recognition, pp. 79–86. Springer (2004)
30. Vedula, S., Baker, S., Rander, P., Collins, R., Kanade, T.: Three-dimensional scene flow. In: Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, vol. 2, pp. 722–729. IEEE (1999)
31. Wiederhold, B., Riva, G.: Balance recovery through virtual stepping exercises using kinect skeleton tracking: a follow-up study with chronic stroke patients. Annual Review of Cybertherapy and Telemedicine 2012: Advanced Technologies in the Behavioral, Social and Neurosciences **181**, 108 (2012)
32. Zabih, R., Woodfill, J.: Non-parametric local transforms for computing visual correspondence. In: Computer Vision?ECCV'94, pp. 151–158. Springer (1994)
33. Zhang, Z.: Iterative point matching for registration of free-form curves and surfaces. International journal of computer vision **13**(2), 119–152 (1994)

10 Appendix

The appendix serves the purpose of showing additional Figures that might be too repetitive for the article as they look somewhat similar.

Fig. 5 These images show the data for the second part of the data set (Squat), and the ground-truth. The images in the first column is the color data and the images in the second column are the depth data. The top image in the third column is the color coded ground truth OF and the bottom image is the flow in the depth direction, where bluer intensities represent movement away from the viewer and vice versa for the redder intensities.

Fig. 6 These images show the data for the third part of the data set (UpDown), and the ground-truth. The images in the first column is the color data and the images in the second column are the depth data. The top image in the third column is the color coded ground truth OF and the bottom image is the flow in the depth direction, where bluer intensities represent movement away from the viewer and vice versa for the redder intensities.

Table 4 Errors on all four parts of the data set. Best scores in each column are bold and the second best score is underlined. The methods appear in the same order as in Table 1, except two results are presented for the first method (without and with color information). The last table is the average score over the four parts of the data set.

Part 1 (UpDown)

Short name	$NRMS_{SF}$ (%)	AAE_{SF} (Degrees)	FDR (%)	$NRMS_{OF}$ (Pixels)	AAE_{OF} (Degrees)
Range Flow depth	7.709	4.734°	21.19	0.2638	4.24°
Range Flow depth and color	6.28	3.03°	91.23	0.2162	2.28°
RGB-D Flow	6.063	4.155°	4.603	**0.1421**	1.795°
aTGV-SF	5.701	4.012°	74.84	0.2135	3.743°
CP-Census	3.84	3.145°	0.4824	0.1682	2.635°
Local/Global	3.604	1.962°	4.306	0.2101	2.459°
Dense Semi-Rigid	**3.59**	**1.768°**	7.124	0.147	**1.665°**
Primal-Dual Flow	5.814	7.734°	0.348	0.2618	7.949°
SHuMo Flow	5.616	3.546°	**0.0039**	0.2472	5.618°

Part 2 (Squat)

Short name	$NRMS_{SF}$ (%)	AAE_{SF} (Degrees)	FDR (%)	$NRMS_{OF}$ (Pixels)	AAE_{OF} (Degrees)
Range Flow depth	17.48	3.746°	11.37	0.8747	3.158°
Range Flow depth and color	10.68	1.265°	16.42	0.4299	0.9842°
RGB-D Flow	7.598	1.431°	1.551	**0.2237**	**0.655°**
aTGV-SF	11.87	1.483°	64.57	0.4747	1.377°
CP-Census	5.73	1.123°	0.653	0.2504	0.784°
Local/Global	**4.63**	1.546°	0.887	0.4091	1.397°
Dense Semi-Rigid	6.558	**1.072°**	4.69	0.2766	0.7212°
Primal-Dual Flow	7.938	8.414°	0.01921	0.6347	10.07°
SHuMo Flow	7.933	6.143°	**0**	0.6242	6.858°

Part 3 (Forward)

Short name	$NRMS_{SF}$ (%)	AAE_{SF} (Degrees)	FDR (%)	$NRMS_{OF}$ (Pixels)	AAE_{OF} (Degrees)
Range Flow depth	16.33	7.149°	90.29	0.5596	7.249°
Range Flow depth and color	5.72	4.849°	91.23	0.3381	4.443°
RGB-D Flow	6.624	4.58°	17.8	**0.2051**	4.493°
aTGV-SF	6.444	5.099°	90.84	0.1846	4.47°
CP-Census	6.32	4.808°	52.85	0.2532	**3.56°**
Local/Global	5.45	5.903°	**16.56**	0.3641	5.606°
Dense Semi-Rigid	**5.231**	5.507°	37.89	0.2536	5.397°
Primal-Dual Flow	6.176	7.722°	16.47	0.3913	8.152°
SHuMo Flow	6.17	**1.281°**	**0.2386**	0.3844	4.437°

Part 4 (Walk)

Short name	$NRMS_{SF}$ (%)	AAE_{SF} (Degrees)	FDR (%)	$NRMS_{OF}$ (Pixels)	AAE_{OF} (Degrees)
Range Flow depth	42.23	3.544°	15.97	1.04	6.715°
Range Flow depth and color	101.3	2.708°	90.18	1.624	9.328°
RGB-D Flow	13.65	1.935°	2.316	**0.3676**	**0.8491°**
aTGV-SF	22.77	1.512°	35.46	0.6181	1.291°
CP-Census	11.16	1.602°	1.145	0.4248	1.277°
Local/Global	**9.627**	**1.439°**	**0.7292**	0.6423	3.282°
Dense Semi-Rigid	12.01	1.787°	2.559	0.4009	1.013°
Primal-Dual Flow	20.32	8.386°	0.01107	0.8774	12.52°
SHuMo Flow	19.92	4.336°	**0**	0.8207	10.34°

Average over the 4 cases

Short name	$NRMS_{SF}$ (%)	AAE_{SF} (Degrees)	FDR (%)	$NRMS_{OF}$ (Pixels)	AAE_{OF} (Degrees)
Range Flow depth	20.94	4.793°	34.7	0.6845	5.342°
Range Flow depth and color	31.09	2.963°	72.27	0.6521	4.259°
RGB-D Flow	8.484	3.025°	6.568	**0.2346**	**1.948°**
aTGV-SF	11.69	3.027°	66.43	0.3727	2.72°
CP-Census	6.763	2.67°	13.78	0.2742	2.064°
Local/Global	**5.827**	2.712°	5.622	0.4064	3.186°
Dense Semi-Rigid	6.847	**2.534°**	13.07	0.2695	2.199°
Primal-Dual Flow	10.06	8.064°	4.212	0.5413	9.672°
SHuMo Flow	9.91	3.826°	**0.06063**	0.5191	6.812°

Fig. 7 These images show the data for the fourth part of the data set (Walk), and the ground-truth. The images in the first column is the color data and the images in the second column are the depth data. The top image in the third column is the color coded ground truth OF and the bottom image is the flow in the depth direction, where bluer intensities represent movement away from the viewer and vice versa for the redder intensities.

Fig. 8 These images show the difference from the methods and the ground-truth for the first part of the data set (Forward). Each row corresponds to a method. The first column is the difference in the x component, the second is the difference in the y component and the last column is the difference in the z component. The methods are the following from top to bottom: (1)-*Local/Global*, (2)-*Dense Semi-Rigid*, (3)-*RGB-D Flow*, (4)-*aTGV-SF*, (5)-*CP-Census*, (6)-*Range Flow depth*, (7)-*Range Flow depth and color*, (8)-*Primal-Dual Flow* and (9)-*SHuMo Flow*. Bluer intensities represent negative values and redder ones represent positive values. The same scale is used in all cases. For visualization purposes, gamma-mapping is applied to the image.

Fig. 9 These images show the difference from the methods and the ground-truth for the second part of the data set (Squat). Each row corresponds to a method. The first column is the difference in the x component, the second is the difference in the y component and the last column is the difference in the z component. The methods are the following from top to bottom: (1)-*Local/Global*, (2)-*Dense Semi-Rigid*, (3)-*RGB-D Flow*, (4)-*aTGV-SF*, (5)-*CP-Census*, (6)-*Range Flow depth*, (7)- *Range Flow depth and color*, (8)-*Primal-Dual Flow* and (9)-*SHuMo Flow*. Bluer intensities represent negative values and redder ones represent positive values. The same scale is used in all cases. For visualization purposes, gamma-mapping is applied to the image.

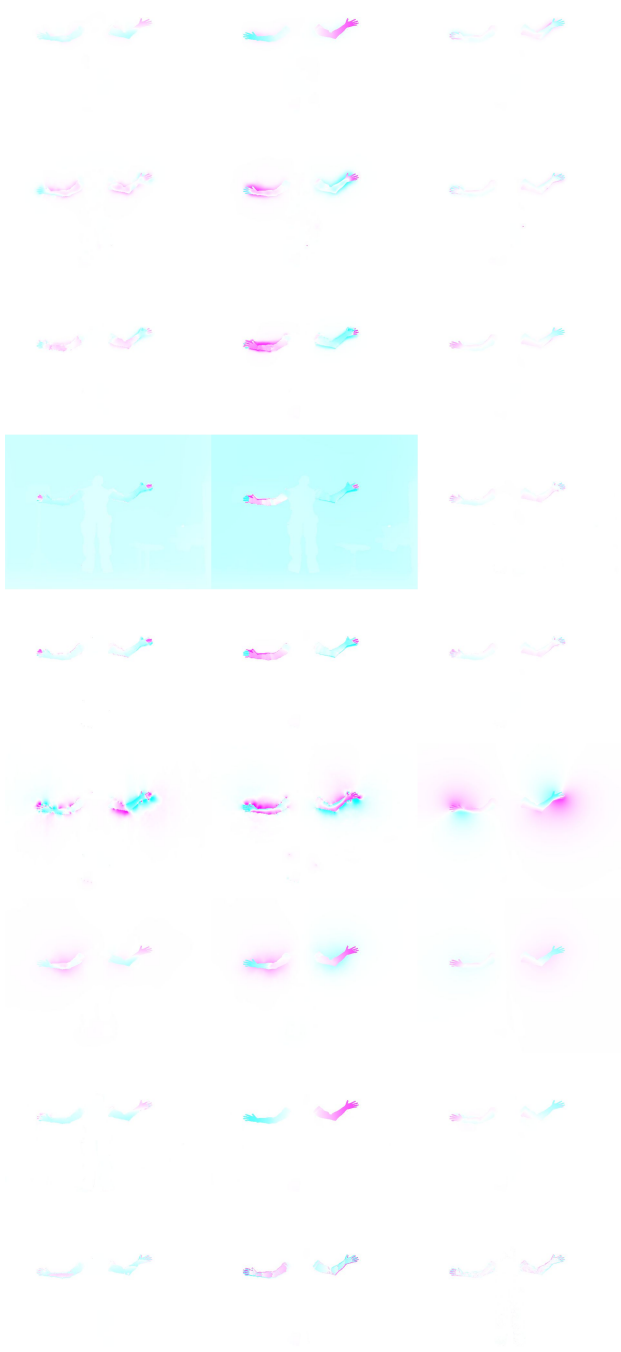


Fig. 10 These images show the difference from the methods and the ground-truth for the third part of the data set (UpDown). Each row corresponds to a method. The first column is the difference in the x component, the second is the difference in the y component and the last column is the difference in the z component. The methods are the following from top to bottom: (1)-*Local/Global*, (2)-*Dense Semi-Rigid*, (3)-*RGB-D Flow*, (4)-*aTGV-SF*, (5)-*CP-Census*, (6)-*Range Flow depth*, (7)- *Range Flow depth and color*, (8)-*Primal-Dual Flow* and (9)-*SHuMo Flow*. Bluer intensities represent negative values and redder ones represent positive values. The same scale is used in all cases. For visualization purposes, gamma-mapping is applied to the image.

Fig. 11 These images show the difference from the methods and the ground-truth for the fourth part of the data set (Walk). Each row corresponds to a method. The first column is the difference in the x component, the second is the difference in the y component and the last column is the difference in the z component. The methods are the following from top to bottom: (1)-*Local/Global*, (2)-*Dense Semi-Rigid*, (3)-*RGB-D Flow*, (4)-*aTGV-SF*, (5)-*CP-Census*, (6)-*Range Flow depth*, (7)- *Range Flow depth and color*, (8)-*Primal-Dual Flow* and (9)-*SHuMo Flow*. Bluer intensities represent negative values and redder ones represent positive values. The same scale is used in all cases. For visualization purposes, gamma-mapping is applied to the image.

Appendix E

Modeling Poses of Infants Using Machine Learning and Motion Tracking

---

In Preparation

Modeling Poses of Infants Using Machine Learning and Motion Tracking

Mikkel Damgaard Olsen[1], Gudmundur Einarsson[1], Jens Bo Nielsen[2,3] and Rasmus R. Paulsen[1]

[1]Department of Applied Mathematics and Computer Science, The Technical University of Denmark
[2]Department of Nutrition, Exercise and Sport Science, University of Copenhagen, Copenhagen, Denmark
[3]Department of Neuroscience and Pharmacology, University of Copenhagen, Copenhagen, Denmark

November 3, 2015

Abstract

Using a dataset of 72 infants recorded with a Microsoft Kinect depth sensor, we analyze the most common poses of infants in the age of 0-6 months corrected age. Using a previously developed system for motion tracking of infants, we quantify the different poses and extract features related to milestones considered in clinical assessment of infant development. We show how variation analysis lead to results closely related to the practical reported movement patterns of infants. These results indicate that motion tracking can be used for assisting clinicians in quantitatively assessing infants' motor development.

Introduction

Even though the movements of a newborn child might seem rather random and chaotic, these movements are important for the infants development, both physically and cognitively. In order to assess the motor development of infants, a number of methods exist, where the doctors/clinicians observe the infant's movements and assess if the infant is able to meet certain motor-specific milestones. This can e.g. be reaching for midline with both hands, moving both arms randomly and varying and reach for knees/feet [14, 12, 9]. Some milestones also involve the infants actively reaching for toys and following an object with its eyes as well as interacting with objects or a person. For obvious reasons, the

different milestones are also divided into groups, based on the age of the infant as well as the position of the infant, i.e. if the infant is in prone, supine, sitting or standing position. The assessment is done by observing the infants either directly or based on video recordings of the infants. This task is time-consuming and as new techniques of recording data arises, methods for assisting the clinicians in the assessment have already been studied [10, 1]. However, the studies usually focus on detecting special kinds of movements, e.g. movements related to the motor disorder cerebral palsy. A more general approach is considered in this work, where an infant motion tracking system is used to obtain pose data of infants and different methods are used to analyze these poses. The idea is that the system can assist doctors detect and diagnose diseases that affects the infant's motor system, such as cerebral palsy.

Human Pose Analysis

Modeling and analyzing poses from motion tracking has been done before on adults, e.g. in [3, 4], where the authors teaches a robot to imitate human motion, using data obtained from a motion tracking system. The authors reduce the dimensionality of the pose data using Principal Component Analysis (PCA) and by fitting loops to the transformed data, a robot is able to learn periodic motions. In [6], the authors model the different poses of humans, by representing the poses using planviews, i.e. synthesizing the data as it was taken from above. Applying PCA to the data, leads to the Eigen-poses represented as heatmaps, that explain the different modes of the poses in the data. PCA is also used in [7], where the authors present a statisical shape and pose model, based on 3D surface scans of adults. Existing studies uses the pose analysis for diagnostic purposes, e.g. in [13], where the authors use motion tracking data to asses postural stability based on assessing the 3 first principal modes from a PCA. In this work, we analyze the pose data from infants and relate it to milestones considered in assessment methods used in practice. A short introduction to the assessment methods and the data is given in the following.

Infant Motor Development

When observing and qualitatively assessing infants' movements and abilities, the different milestones are divided into age-groups (in months). In the early months, an infant will not be able to sit or stand and the milestones are constrained to movements in supine and prone positions as well as positions where the infant is held by a person. In the early months, the infant's ability to use the different limbs is in focus, more than the ability to control the limbs. When the infant gets older, it gets better control of its movements and becomes aware of its own body. This yields to movements related to reaching for objects and the infant begins to reach for its own feet and knees. Next, when the infant is strong enough, it is able to roll from supine to prone and back again and later the infant is able to crawl, stand and walk.

Table 1: Different assessment methods are currently used for assessing the infants development. The methods focus on different age groups as well as different motor and cognitive tasks.

Name	Age Months	Time Required Minutes	Testing
Bayley	1-42	30-90	motor, cognitive and language development
AIMS	0-18 or until walking	20-30	motor development
IMP	3-18	15	motor development
HINT	2-13	15-30	motor and cognitive development

A numerous list of methods exists for assessing the infants development and some of them are mentioned below, where the names are the abbreviations for the respective methods. The different tests are summarized in Table 1.

- Bayley: Bayleys Scale of Infant Development [2]
- AIMS: Alberta Infant Motor Scale [12]
- IMP: Infant Motor Profile [9]
- HINT: Harris Infant Neuromotor Test [5]

Common to all of the methods, is that they are based on observing and scoring the movements of infants in relation to a predefined set of expected movements and/or poses. It should be noted that the physicians experience might bias the results and the techniques usually requires participation on specific courses. An objective approach for assessing infants and assist the clinicians, is thus desired. This study focus on such a system, where we objectively consider the primary poses of infants, using statistical methods. The knowledge gained from this analysis is later used for improving the used motion tracking system, as the estimation can be initialized by the most likely poses. Furthermore, we demonstrate how this system can be used in practice by defining measures of different milestones.

Data

72 infants in the age of 0-30 weeks were included in this study. The age is corrected with respect to term, i.e. at term, the age of an infant is 0 weeks, even though the infant might be born pre or post-term. The data has been

Figure 1: Example of the data available for each frame in the data. Left image shows the color image while the right image shows the depth data in millimeters visualized as a grayscale image. Dark colors indicate that an object is closer to the camera, compared to brighter objects, which indicate objects further away. Completely dark pixels are occluded regions, due to the technicalities of the depth estimation.

recorded with a Microsoft Kinect Sensor, but any other depth sensor can be used instead. The data consist of both color data, similar to data obtained by ordinary cameras and depth data, that represent the distance from the sensor to the observed scene for each frame (see Figure 1). Each recording consists of 5-10 minutes of data, capturing the voluntary movements of infants in supine position at a framerate of 30 images per second. During the recordings, the infants were not stimulated by sound or visual inputs. Neither did the infants have pacifiers or toys that would effect their movements. The protocol used during the recordings is summarized here:

- External Stimulation: No sounds or visual stimulation was used during the recordings. In addition the recordings were done in a quiet room and all persons were placed out of sight of the infant.
- Clothing: In order to capture the movements of the outer extremities, the infants did not wear socks or rompers that would otherwise hinder observing movements in the joints. Some infants wore a short-sleeved bodystocking and in the case of a long-sleeved bodystocking, the sleeves were pulled up as much as possible, while ensuring that the infant was able to move freely.
- Position: The infants were positioned in supine position on a mattress on the floor.
- Infant Conditions: The infants were not sleepy or hungry and in case of crying the recordings were stopped. In addition, the parents were informed that the recordings could be stopped at any point.

The recordings are summarized in Figure 2, where the infants have been divided into groups based on age in months. In addition a plot illustrates the

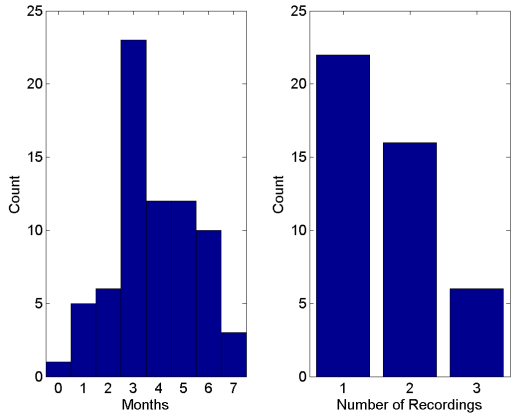


Figure 2: Left: Age of infant recordings in the data set. Right: Summary of repeated recordings of the same infants.

number of recordings for the infants, where the timespan between two recordings is approximately one month. Infants are not resampled, i.e. an infant with 3 recordings is not included as an infant with 1 and 2 recordings.

1 Methods

Motion Tracking

The motion tracking is done using a previously published method for tracking motions of infants [11]. The method is based on an articulated 3D model that represents the surface of the infant. A set of parameters define the pose of the infant by means of relative joint angles between connected body parts. For each frame, the motion tracking estimates a state vector, explaining the joint angles as well as the global position, orientation and size. The motion tracking of a full video consisting of n frames, can thus be represented in a single matrix $\mathbf{\Theta}$, where each column is a state vector described by 62 parameters.

$$\mathbf{\Theta} = [\theta_1 \dots \theta_f \dots \theta_n] \tag{1}$$

Given a state vector, the 3D positions of the joints can be obtained, by use of forward kinematics. The modeled joint positions are; Bodycenter, Neck, Headcenter, Shoulders, Elbows, Hands, Crotch, Hips, Knees, Ankles and Toes. In this work, we focus on the spatial features rather than the angular features, as the pose is simplest visualized as the set of positions in space. One example

of the resulting pose of and infant can be seen in Figure 3. However, for the remained of the study, a simpler stickman figure will be used for visualizing the poses. The stickman figure related to the pose in Figure 3, can be seen in Figure 4.

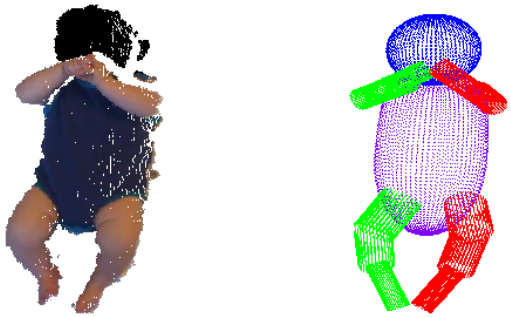


Figure 3: The resulting pose estimated for one frame in the dataset. The model is shifted for visualization purposes.

Concatenating all joint positions from the infants in one data matrix with n observations, where $n \approx [10000 \text{ frames} \cdot \text{Number of Infants}]$, leads to the input data for the later analysis. The number of variables is based on the number of joints included. In this work, the above mentioned joints are used, which lead to 33 variables (3 for each joint). As we are only interested in the poses of the infants and not the variation in shape/size, the positions are normalized with respect to the global position, orientation and size.

Clustering and Variance Analysis

In order to extract the most common poses, different techniques from machine learning are considered. In machine learning, one way to find similar structures in data, is by use of clustering, where observations are clustered into groups based the measured variables [8]. The goal is to cluster the data such that observations belonging to the same group, lie close to each other, while observations belonging to different groups lie far from each other. In order to measure how far or close two observations are, some metric should be defined beforehand, but usually the Euclidean distance is used. Other metrics can be used, such as the Mahalanobis distance, in order to incorporate the (co-)variance of the variables. Another thing that should be considered, is the scaling of the variables, which might be different. Once an appropriate metric has been defined, the number of clusters should also be considered. The number of clusters should be much smaller than the number of observations, but still big enough to group similar observations while separating dissimilar observations. Different

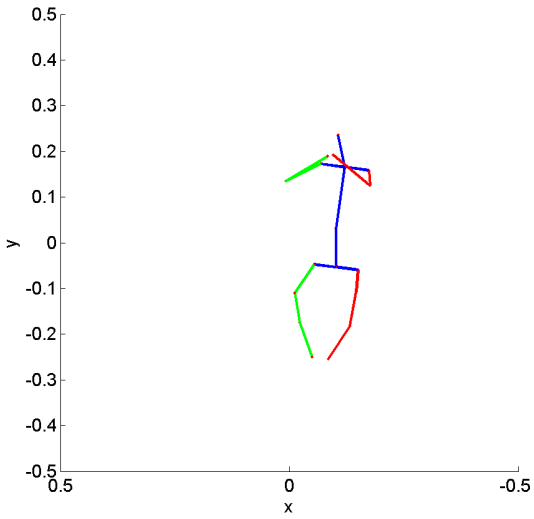


Figure 4: A simpler stickman figure is used to visualize the pose, compared to the 3D model shown in Figure 3.

techniques exists for choosing the number of clusters or even change the number of clusters, during the clustering, based on different model criteria. The different choices results in different clustering methods. However, in this work, we use the popular K-means clustering, where the number of clusters K is chosen beforehand. This methods iteratively updates the cluster centers as well as the labels of the different observations.

The clustering approach is able to find the most frequent poses and simultaneously label each observation as one the estimated cluster centers. However, clustering will not explain how different body parts covariate. It is likely that there might be some correlation between body parts, e.g. in the sense of symmetry between the left and right side of the body. The concept of factor analysis and dimension reduction is often used for this task, namely by use of the well known and popular method; Principal Component Analysis. The idea with this approach is to transform the variables in such a way that, the bases of the transformation, also called principal components, point in the directions of maximum variance. This is often used for dimension reduction, where two originally correlated variables can be explained by a single transformed variable. Using the PCA closely relates the task to statistical shape modeling, even though we consider poses instead of shapes. It should be noted that other methods exist

for dimension reduction, but in this work we focus on PCA.

2 Results

In the following, the results from the clustering and variance analysis is shown. The data has been divided into three subsets, based on the age of the infants. This is done in order to see how results change with age.

Pose Clustering

Clustering the pose-data for different ages, results in an overview of the most common poses in the dataset, with respect to the age of the infants. Using a simple K-means clustering approach, with $K = 20$, we use the cluster-centers as the most frequent poses in the data. The results are illustrated in Figure 5, where it can be seen that older infants tend to have more poses with their legs lifted, from the ground, i.e. anti-gravity movements/poses. Moreover, hands together or hands touching or reaching for the feet/knees are also more frequent in the older infants. For the younger infants, the most frequent poses describe different orientations of the extremities, but without a specific goal in mind.

We also cluster the poses with respect to the upper and lower regions of the body. This is done in Figure 6 and Figure 6, respectively. For the upper region, i.e. head, arms and upper body, there is no obvious symmetry for the younger infants, while this is more apparent in the older infants. For the older infant, the hand-to-hand and hand-to-feet poses are also seen in the clusters. For the lower body region, there is less variation between the different age groups, indicating that most pose variation happens in the upper body.

Pose Variation

In addition to the clustering of the poses, we analyze how the different limbs covariate using PCA. Figure 8 illustrates the samples along the first principal component, based on different ages. It can be observed that for the young infants, the principal component models how "open" the infants are with both arms and legs. This behavior is also modeled in the first principal component of the older infants, but the infants tend to have more goal-specific poses which are seen by the configuration of the hands and legs. For the three age groups, the first principal component explains 19, 23 and 37 percent of the total variation, ordered from young to old. The reason for this decrease in randomness, might be the fact that older infants have more control of their movements and their poses are more related to certain goal-specific tasks. The younger infants has not yet learned to control their movements and the movements are dominated by spontaneous movements, which tend to be more random.

As in the clustering, the upper and lower body regions are considered as well. The results can be seen in Figure 9 and 10, respectively. Even though the upper/lower results does not deviate a lot from the total body results in

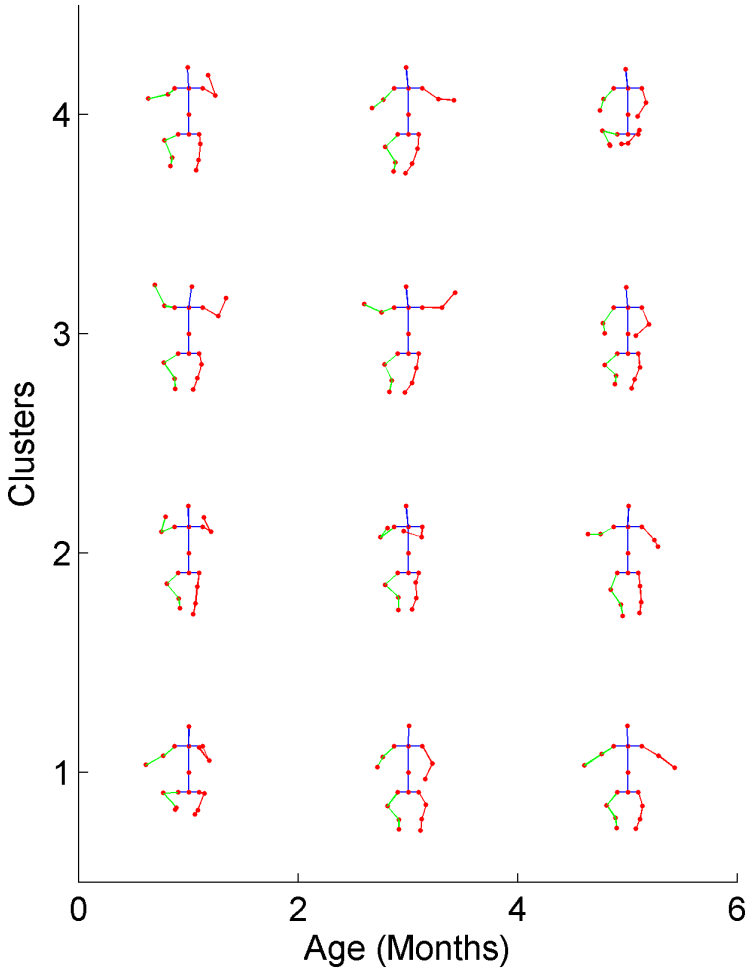


Figure 5: Clustering the pose-data for different ages, gives an overview of the most common poses in the dataset, as a function of age. The cluster-centers are ordered in decreasing cluster size.

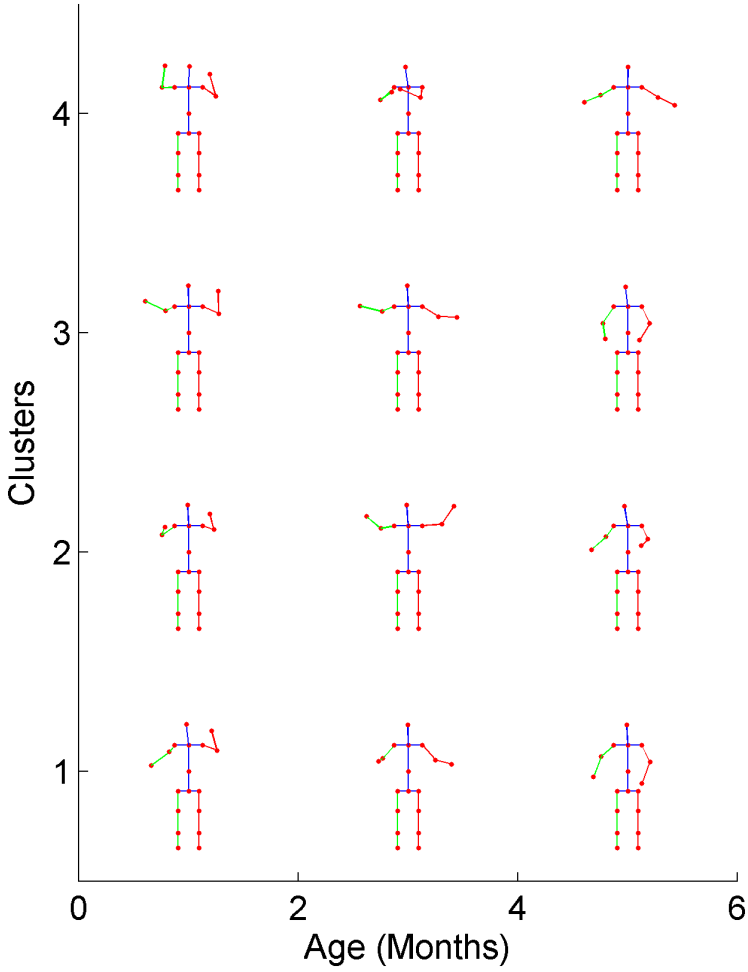


Figure 6: Clusters only based on the joints of the head, arms and upper body.

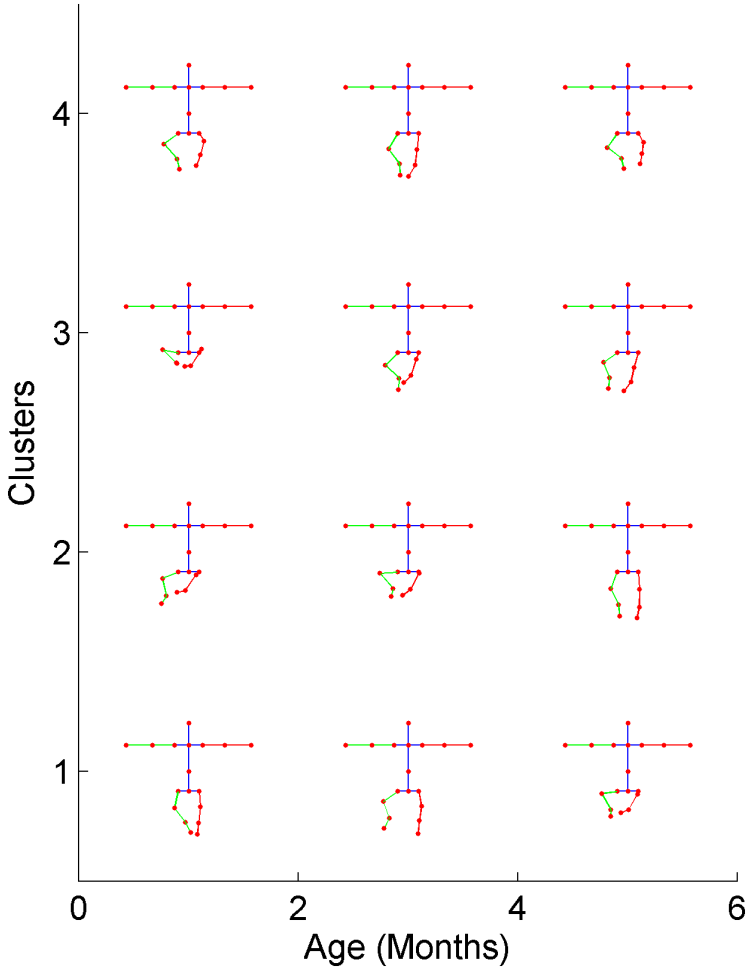


Figure 7: Clusters only based on the joints of the lower body and lower extremities.

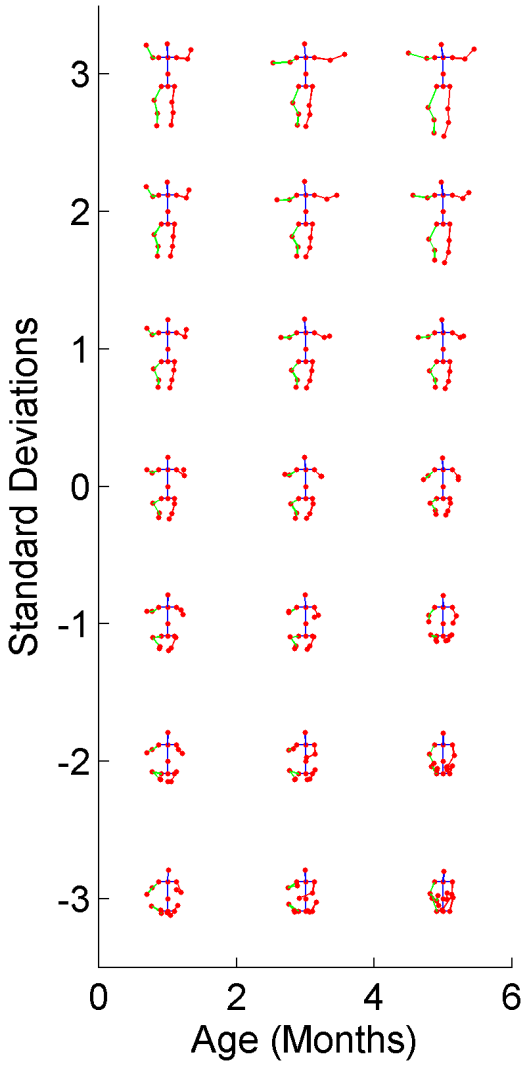


Figure 8: The variation of the poses for different ages, sampled along the first principal component.

Figure 8, it is easier to see the age-dependent tasks of the hands, as well as the anti-gravity poses in the lower extremities.

Prior Based Pose Estimation

Given the current database of infants with associated estimated poses, we have an idea of the most common poses of infants. This can be used as prior information for pose estimation of new infants, as some poses are more probable than others, as seen in the previous analysis. The data for the new infant can thus be tested against a number of candidate poses. The candidate pose can e.g. be three different configurations of the arms; both arms point up, both arms point down or arms pointing in opposite directions. This is only a simple example, but the different configurations of the legs should be included as well. For an infant not included in the pose clustering, the resulting poses for 4 frames are illustrated in Figure 11. As seen, most body parts are correctly aligned by the pose estimation, but with some deviations in relation to the correct orientation of specially the outer limbs. This is due to the fact that all variation is not included in the candidate poses. One way to improve the result is to increase the number of candidate-poses, at the cost of speed. Another solution could assume that the age is known beforehand and the set of candidate poses are limited to this age group.

Assessing Motor Milestones

Using the spatial position of different body parts, the system is able to tell if an infant is bringing its hands together, doing anti-gravity movements, reaching for its feet or knees, etc. These milestones can be quantified by features generated from the motion tracking results. The quantities listed below are just a subset of the possible quantities that can be extracted from the results.

- Motions in both sides at the same time: This quantity can be measured as the correlation between two opposite limbs, based on the velocities of the limbs. The parameter considers both magnitude and direction of the movements and measures the correlations between the two sides.
- Anti-gravity with legs: This quantity is measured using two features. One is the angle between the thigh and the ground and the other feature is simply based on the projected distance from the knee or foot, to the ground surface.
- Feet to mouth: Measured as the distance between the left or right foot to the head bodypart.
- Hands to feet: Measured as the Euclidean distance between one hand and one foot. This can both include same side touchings as well as crossings, where the left hand reaches for the right leg.

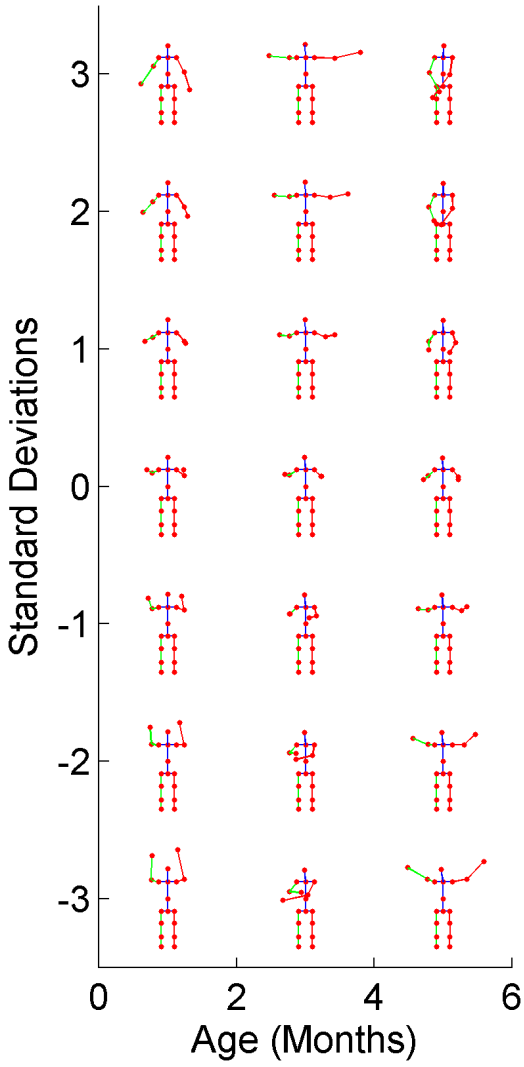


Figure 9: The variation of the upper body region for different ages, sampled along the first principal component.

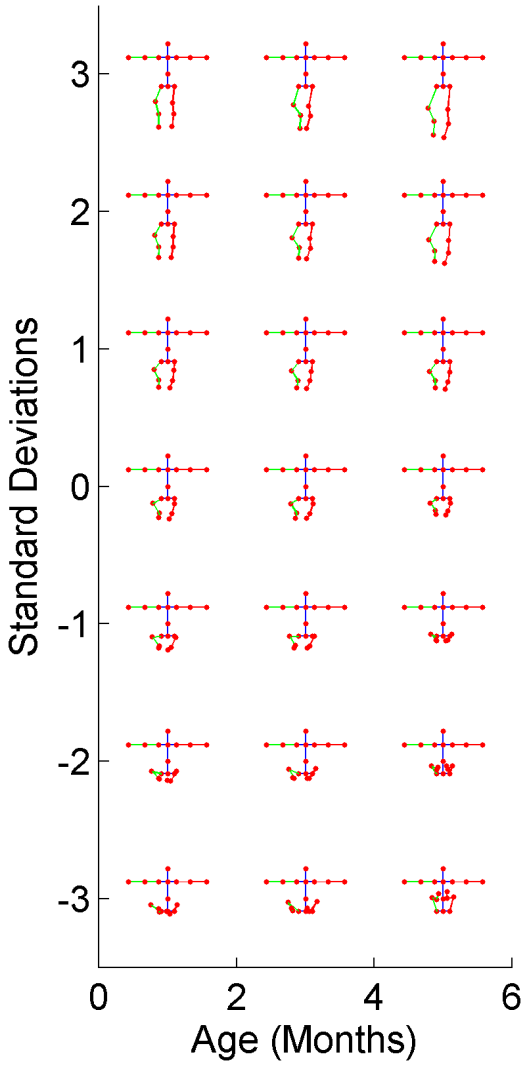


Figure 10: The variation of the lower body region for different ages, sampled along the first principal component.

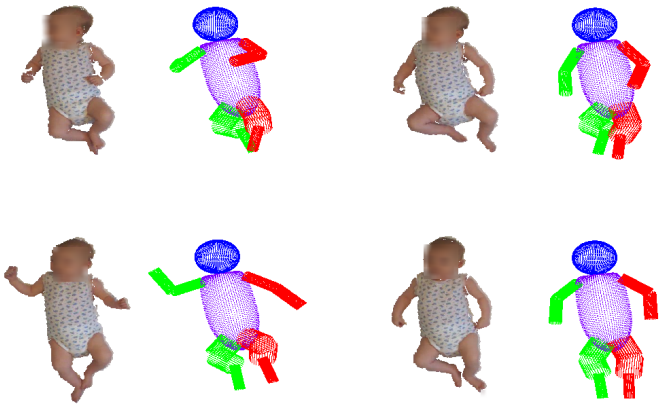


Figure 11: Based on the learned poses, we estimate the pose of an infant excluded from the dataset. The resulting pose estimation can be seen for 4 frames. As seen, the overall pose is captured, but details such as the orientation of the feet is not correct.

- Hands to mouth: Measured as the distance between the left or right hand to the head bodypart.
- Hands together: Measured as the distance between the left and right hand.

Classifying each frame as containing the different milestones or not, we can count the number of occurrences of each milestone, divided into the different age groups. The counting is done such that neighboring frames are combined into one occurrence, if the milestone is found in both frames. This is done to remove redundant poses, where two successive frames most likely contains similar poses. Threshold parameters, i.e. distance, velocity and angular parameters, are chosen manually, based on the unit of the metric used for the respective milestone. The average occurrence of the different milestones are summarized in Figure 12, divided into the different age groups. The occurrences are normalized to one minute, such that the different age groups can be compared and it can be observed that as for the older infants, goal-specific goals such as reaching and rolling increases. Especially for the hand-feet milestone, there is a big increase of occurrences. This number of occurences is of course based on the user-defined thresholds. However, in Figure 13 the mean distance between the hands and the feet are visualized for each infant, as a function of corrected age in weeks. From this it is obvious that the older infants indeed tend to have smaller distances between hands and feet. Some of the older infants do not

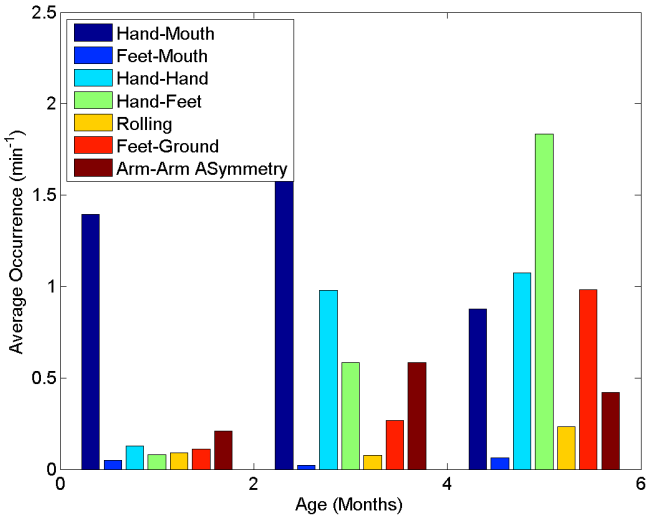


Figure 12: Based on the definition of the milestones, the occurrence in the different age groups are summarized. The occurrences are normalized to one minute.

have these low distances, but this can of course still be considered as normal.

Conclusion

Based on 72 recordings of infants in the age of 0-6 months, we discovered how the poses and movements of infants vary with age. This was found using statistical methods on data obtained from a newly developed infant motion tracking system. The statistical results was based on finding the most frequent poses in infants and the results clearly related to the expected variation and poses in different age groups. Furthermore, the system demonstrated how different motor related milestones could be quantified and used for automatic detection. The features and milestones considered in this study are only chosen to demonstrate the usefulness of using motion tracking as a tool for assessing infant motor development. Other relevant parameters might be extracted as well and used for describing the pose and motion of infants at different stages through the early months of life. For future work, the analysis can be applied in motion-space in order to model the most common motions of infants. The input data to the different methods would not only be single poses, but also changes be-

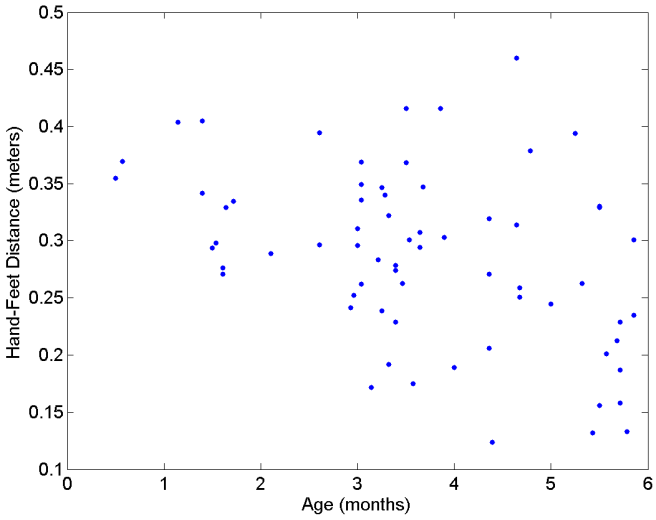


Figure 13: For the hand-feet milestone, the mean distance for each infant is visualized. It is seen that older infants tend to reach more for their feet.

tween the poses. However, based on the goal with this approach, some sort of normalization or mapping should be considered, as different starting poses will result in different movements, even though the movement might seem similar. Furthermore, it might be necessary to remove frames which contain too little movement. In addition to the described analysis, future work can focus on combining the motion data with the pathological history of the infants, when they get older. This might lead to quantitative classification of diseases related to motion development.

A Appendix

This section contains sampled poses along the second and third principal component, for the total, upper and lower poses, respectively.

References

[1] Lars Adde, Jorunn L. Helbostad, Alexander Refsum Jensenius, Gunnar Taraldsen, and Ragnhild Støen. Using computer-based video analysis in

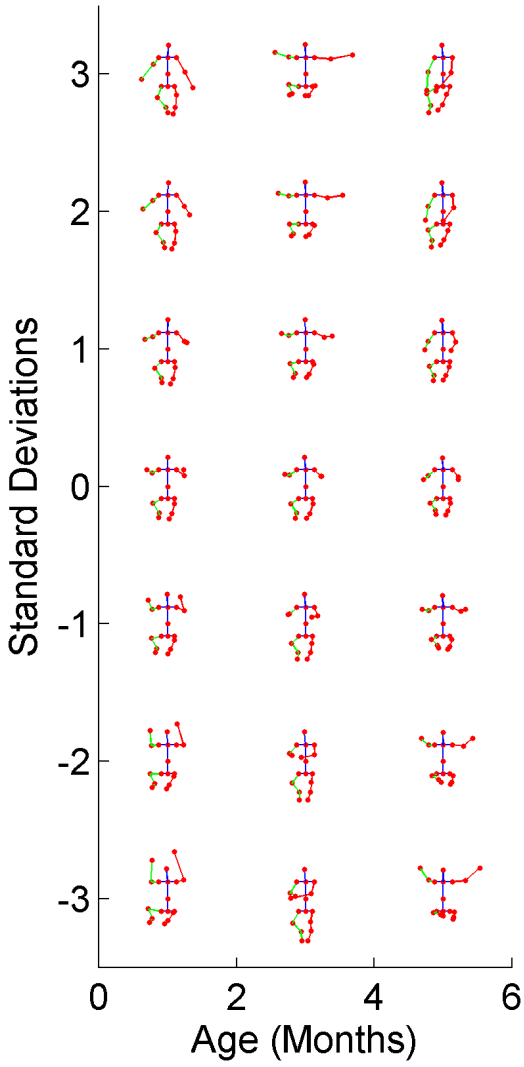


Figure 14: The variation of the poses for different ages, sampled along the second principal component.

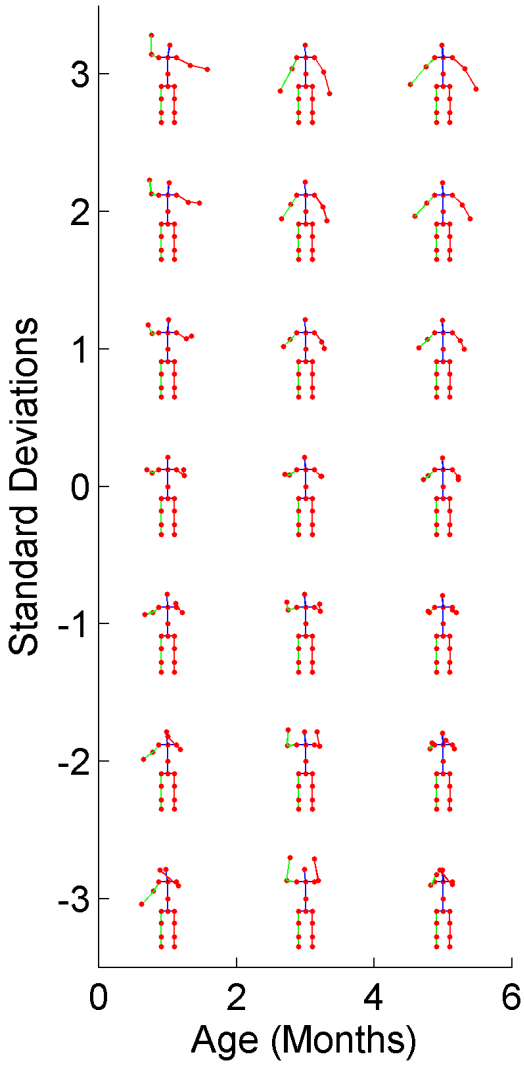


Figure 15: The variation of the upper body region for different ages, sampled along the second principal component.

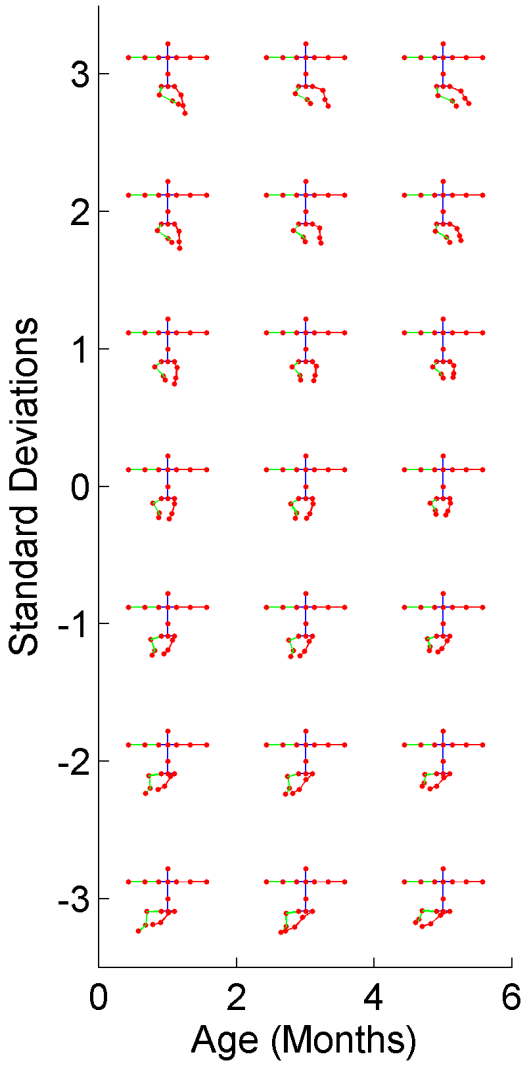


Figure 16: The variation of the lower body region for different ages, sampled along the second principal component.

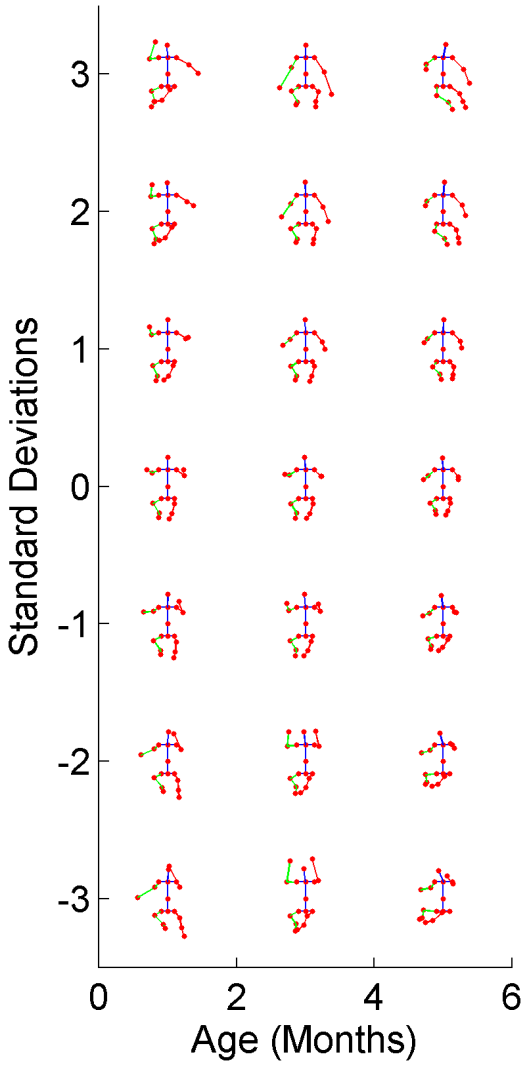


Figure 17: The variation of the poses for different ages, sampled along the third principal component.

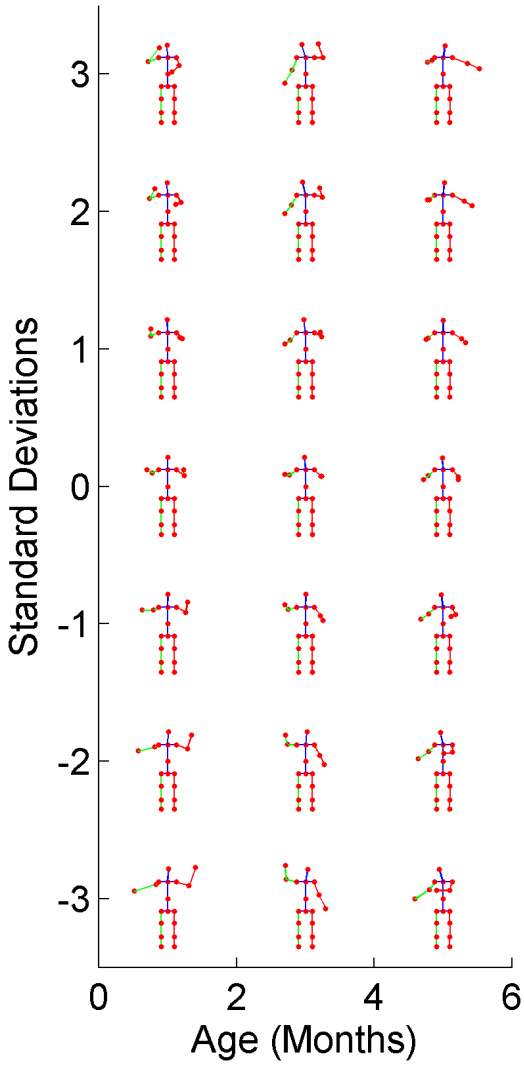


Figure 18: The variation of the upper body region for different ages, sampled along the third principal component.

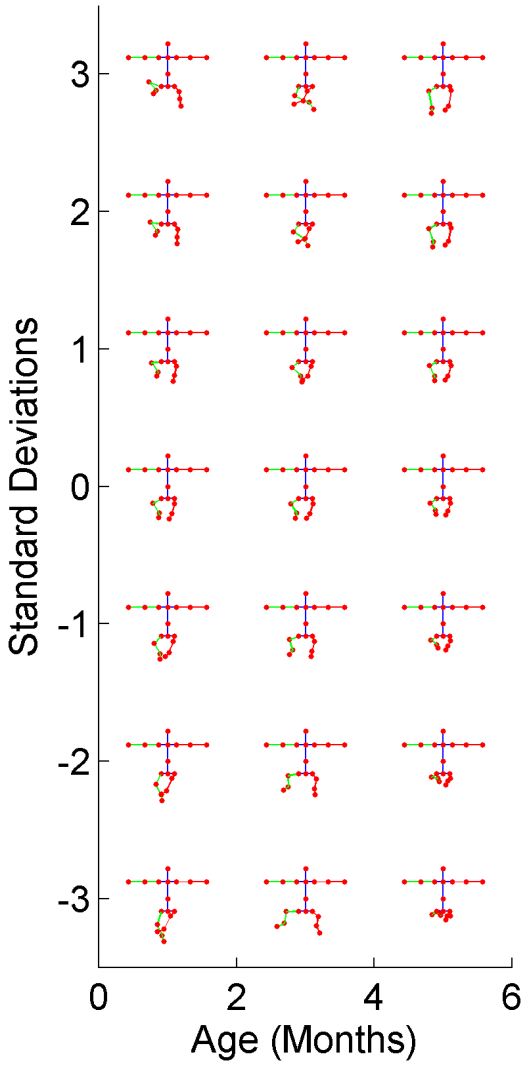


Figure 19: The variation of the lower body region for different ages, sampled along the third principal component.

the study of fidgety movements. *Early Human Development*, 85(9):541–547, 2009.

[2] N. Bayley. *Manual for the Bayley Scales of Infant Development*. Psychological Corporation, 1969.

[3] R. Chalodhorn and R.P.N. Rao. Using eigenposes for lossless periodic human motion imitation. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 2502–2509, Oct 2009.

[4] Rawichote Chalodhorn and Rajesh P. N. Rao. Learning to imitate human actions through eigenposes. In Olivier Sigaud and Jan Peters 0001, editors, *From Motor Learning to Interaction Learning in Robots*, volume 264 of *Studies in Computational Intelligence*, pages 357–381. Springer, 2010.

[5] Susan R. Harris, Antoinette M. Megens, Catherine L. Backman, and Virginia Hayes. Development and standardization of the harris infant neuromotor test. *Infants & Young Children*, 16(2):143–151, 2003.

[6] Michael Harville and Dalong Li. Fast, integrated person tracking and activity recognition with plan-view templates from a single stereo camera. In *CVPR (2)*, pages 398–405, 2004.

[7] Nils Hasler, Carsten Stoll, Martin Sunkel, Bodo Rosenhahn, and Hans-Peter Seidel. A statistical model of human pose and body shape. *Comput. Graph. Forum*, 28(2):337–346, 2009.

[8] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

[9] Kirsten R Heineman, Arend F Bos, and Mijna Hadders-Algra. The infant motor profile: a standardized and qualitative method to assess motor behaviour in infancy. *Developmental Medicine & Child Neurology*, 50(4):275–282, apr 2008.

[10] Dominik Karch, Keun-Sun Kim, Katarzyna Wochner, Joachim Pietz, Hartmut Dickhaus, and Heike Philippi. Quantification of the segmental kinematics of spontaneous infant movements. *Journal of Biomechanics*, 41(13):2860–2867, sep 2008.

[11] Mikkel Damgaard Olsen, Anna Herskind, Jens Bo Nielsen, and Rasmus Reinhold Paulsen. Model-based motion tracking of infants. In *Computer Vision - ECCV 2014 Workshops*, pages 673–685. Springer International Publishing, 2015.

[12] M.C. Piper and J. Darrah. *Motor Assessment of the Developing Infant*. Saunders, 1994.

[13] Joseph D. Skufca, Erik M. Bollt, Rakesh Pilkar, and Charles J. Robinson. Eigenposes: Using principal components to describe body configuration for analysis of postural control dynamics. In *IJCNN*, pages 1–3. IEEE, 2010.

[14] Charlene L. Stokamer, Arlene Eisenberg, Heidi E. Murkoff, and Sandee E. Hathaway. What to expect the first year. *The American Journal of Nursing*, 90(2):104, feb 1990.

Bibliography

[1] Accardo, J., Kammann, H., and Jr, A. H. H. Neuroimaging in cerebral palsy. *The Journal of Pediatrics 145*, 2 (aug 2004), S19–S27.

[2] Adde, L. *Prediction of cerebral palsy in young infants : Computer-based assessment of general movements.* PhD thesis, Norwegian University of Science and Technology, Department of Laboratory Medicine, Children's and Women's Health, 2010.

[3] Adde, L., Helbostad, J., Jensenius, A. R., Langaas, M., and Støen, R. Identification of fidgety movements and prediction of CP by the use of computer-based video analysis is more accurate when based on two video recordings. *Physiotherapy Theory and Practice 29*, 6 (aug 2013), 469–475.

[4] Adde, L., Helbostad, J. L., Jensenius, A. R., Taraldsen, G., and Støen, R. Using computer-based video analysis in the study of fidgety movements. *Early Human Development 85*, 9 (sep 2009), 541–547.

[5] Agarwal, A., and Triggs, B. Recovering 3d human pose from monocular images. *IEEE Trans. Pattern Anal. Mach. Intell. 28*, 1 (Jan. 2006), 44–58.

[6] Andersen, M., Jensen, T., Lisouski, P., Mortensen, A., Hansen, M., Gregersen, T., and Ahrendt, P. *Kinect Depth Sensor Evaluation for Computer Vision Applications.* Technical report. Aarhus University, Department of Engineering, 2012.

[7] Baak, A., Müller, M., Bharaj, G., Seidel, H.-P., and Theobalt, C. A data-driven approach for real-time full body pose reconstruction

from a depth camera. In *IEEE 13th International Conference on Computer Vision (ICCV)* (Nov. 2011), IEEE, pp. 1092–1099.

[8] Bax, M., Goldstein, M., Rosenbaum, P., Leviton, A., Paneth, N., Dan, B., Jacobsson, B., and Damiano, D. Proposed definition and classification of cerebral palsy, april 2005. *Developmental Medicine & Child Neurology null* (8 2005), 571–576.

[9] Berg, A. Modellbasert klassifisering av spedbarns bevegelser, 2008.

[10] Bilde, P. E., Kliim-Due, M., Rasmussen, B., Petersen, L. Z., Petersen, T. H., and Nielsen, J. B. Individualized, home-based interactive training of cerebral palsy children delivered through the internet. *BMC Neurology 11*, 1 (2011), 32.

[11] Brubaker, M. A., Sigal, L., and Fleet, D. J. Physics-based human motion modeling for people tracking: A short tutorial, 2009.

[12] Chen, Y.-P., and Howard, A. M. Effects of robotic therapy on upper-extremity function in children with cerebral palsy: A systematic review. *Developmental Neurorehabilitation* (aug 2015), 1–8.

[13] Dalal, N., and Triggs, B. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* (2005), Institute of Electrical & Electronics Engineers (IEEE).

[14] Das, S., Trutoiu, L., Murai, A., Alcindor, D., Oh, M., la Torre, F. D., and Hodgins, J. Quantitative measurement of motor symptoms in parkinson's disease: A study with full-body motion capture data. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (aug 2011), Institute of Electrical & Electronics Engineers (IEEE).

[15] de Vries, J., Visser, G., and Prechtl, H. The emergence of fetal behaviour. i. qualitative aspects. *Early Human Development 7*, 4 (1982), 301 – 322.

[16] Dijkstra, E. W. A note on two problems in connexion with graphs. *NUMERISCHE MATHEMATIK 1*, 1 (1959), 269–271.

[17] Droeschel, D., and Behnke, S. 3d body pose estimation using an adaptive person model for articulated icp. In *Proceedings of the 4th International Conference on Intelligent Robotics and Applications - Volume Part II* (Berlin, Heidelberg, 2011), ICIRA'11, Springer-Verlag, pp. 157–167.

[18] Einspieler, C., Prechtl, H., Bos, A., Ferrari, F., and Cioni, G. *Prechtl's Method on the Qualitative Assessment of General Movements in Preterm, Term and Young Infants.* Clinics in Developmental Medicine. Wiley, 2008.

[19] Enzweiler, M., and Gavrila, D. Monocular pedestrian detection: Survey and experiments. *IEEE Transactions on Pattern Analysis and Machine Intelligence 31*, 12 (dec 2009), 2179–2195.

[20] Ferrari, F., Cioni, G., Einspieler, C., Roversi, M. F., Bos, A. F., Paolicelli, P. B., Ranzi, A., and Prechtl, H. F. R. Cramped synchronized general movements in preterm infants as an early marker for cerebral palsy. *Arch Pediatr Adolesc Med 156*, 5 (may 2002), 460.

[21] Foxlin, E., and Inc, I. Chapter 7. motion tracking requirements and technologies.

[22] Gade, R., Jørgensen, A., and Moeslund, T. B. Occupancy analysis of sports arenas using thermal imaging. In *VISAPP 2012 - Proceedings of the International Conference on Computer Vision Theory and Applications, Volume 2, Rome, Italy, 24-26 February, 2012.* (2012), pp. 277–283.

[23] Gagalowicz, A., and Quah, C. K. 3d model-based marker-less human motion tracking in cluttered environment. In *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops* (sep 2009), Institute of Electrical & Electronics Engineers (IEEE).

[24] Ganapathi, V., Plagemann, C., Koller, D., and Thrun, S. Real-time human pose tracking from range data. In *Proceedings of the European Conference on Computer Vision (ECCV)* (2012).

[25] Gough, M., Fairhurst, C., and Shortland, A. Botulinum toxin and cerebral palsy: time for reflection? *Dev Med Child Neurol 47*, 10 (sep 2005), 709.

[26] Hadders-Algra, M. Early diagnosis and early intervention in cerebral palsy. *Frontiers in Neurology 5* (sep 2014).

[27] Hastie, T., Tibshirani, R., and Friedman, J. *The Elements of Statistical Learning.* Springer New York, 2009.

[28] Hauberg, S., Sommer, S., and Pedersen, K. S. Gaussian-like spatial priors for articulated tracking. In *Proceedings of the 11th European conference on Computer vision: Part I* (Berlin, Heidelberg, 2010), ECCV'10, Springer-Verlag, pp. 425–437.

[29] Herskind, A., Greisen, G., and Nielsen, J. B. Early identification and intervention in cerebral palsy. *Dev Med Child Neurol 57*, 1 (jul 2014), 29–36.

[30] Himmelmann, K. Epidemiology of cerebral palsy. *Handbook of Clinical Neurology 111* (2013), 163–167.

[31] Jean, J. S. *Kinect Hacks: Tips & Tools for Motion and Pattern Detection*, 1st ed. O'Reilly Media, Inc., 2012.

[32] Kanemaru, N., Watanabe, H., Kihara, H., Nakano, H., Takaya, R., Nakamura, T., Nakano, J., Taga, G., and Konishi, Y. Specific characteristics of spontaneous movements in preterm infants at term age are associated with developmental delays at age 3 years. *Dev Med Child Neurol* (apr 2013), n/a–n/a.

[33] Karayiannis, N. B., Varughese, B., Tao, G., Jr., J. D. F., Wise, M. S., and Mizrahi, E. M. Quantifying motion in video recordings of neonatal seizures by regularized optical flow methods. *IEEE Transactions on Image Processing 14*, 7 (2005), 890–903.

[34] Karch, D., Kang, K.-S., Wochner, K., Philippi, H., Hadders-Algra, M., Pietz, J., and Dickhaus, H. Kinematic assessment of stereotypy in spontaneous movements in infants. *Gait & Posture 36*, 2 (jun 2012), 307–311.

[35] Karch, D., Kim, K.-S., Wochner, K., Pietz, J., Dickhaus, H., and Philippi, H. Quantification of the segmental kinematics of spontaneous infant movements. *Journal of Biomechanics 41*, 13 (sep 2008), 2860–2867.

[36] Karch, D., Wochner, K., Kim, K., Philippi, H., Pietz, J., and Dickhaus, H. Detection of complex movement patterns in multivariate kinematic time series for diagnostics in pediatric neurology. In *IFMBE Proceedings*. Springer Science + Business Media, 2009, pp. 771–774.

[37] Kato, Z., and Zerubia, J. *Markov random fields in image segmentation*. Foundations and Trends in Signal Processing. Now Publishers, Sept. 2012. 164 pages.

[38] Keskin, C., Kirac, F., Kara, Y., and Akarun, L. Real time hand pose estimation using depth sensors. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on* (2011), pp. 1228–1234.

[39] Khoshelham, K., and Elberink, S. O. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors 12*, 12 (feb 2012), 1437–1454.

[40] Larsen, A. B. L., Hauberg, S., and Pedersen, K. S. Unscented kalman filtering for articulated human tracking. In *SCIA* (2011), A. Heyden and F. Kahl, Eds., vol. 6688 of *Lecture Notes in Computer Science*, Springer, pp. 228–237.

[41] Lee, J. C. Hacking the nintendo wii remote. *IEEE Pervasive Computing 7*, 3 (July 2008), 39–45.

[42] Li, Z., and Kulic, D. Particle filter based human motion tracking. In *2010 11th International Conference on Control Automation Robotics & Vision* (dec 2010), Institute of Electrical & Electronics Engineers (IEEE).

[43] Ly, D. L., Saxena, A., and Lipson, H. Pose estimation from a single depth image for arbitrary kinematic skeletons. *Computing Research Repository* (2011).

[44] Mankoff, K. D., and Russo, T. A. The kinect: a low-cost, high-resolution, short-range 3d camera. *Earth Surf. Process. Landforms 38*, 9 (jul 2013), 926–936.

[45] Meinecke, L., Breitbach-Faller, N., Bartz, C., Damen, R., Rau, G., and Disselhorst-Klug, C. Movement analysis in the early detection of newborns at risk for developing spasticity due to infantile cerebral palsy. *Human movement science 25*, 2 (Apr. 2006), 125–144.

[46] Micilotta, A. S., Ong, E. J., and Bowden, R. Detection and tracking of humans by probabilistic body part assembly. In *Procedings of the British Machine Vision Conference 2005* (2005), British Machine Vision Association and Society for Pattern Recognition.

[47] Monnier, C., German, S., and Ost, A. A multi-scale boosted detector for efficient and robust gesture recognition. In *Computer Vision - ECCV 2014 Workshops*. Springer Science + Business Media, 2015, pp. 491–502.

[48] Murphy, N., and Such-Neibar, T. Cerebral palsy diagnosis and management: the state of the art. *Current Problems in Pediatric and Adolescent Health Care 33*, 5 (may 2003), 146–169.

[49] Neverova, N., Wolf, C., Taylor, G. W., and Nebout, F. Multi-scale deep learning for gesture detection and localization. In *Computer Vision - ECCV 2014 Workshops*. Springer Science + Business Media, 2015, pp. 474–490.

[50] Ni, B., Kassim, A. A., and Winkler, S. A hybrid framework for 3-d human motion tracking. *IEEE Trans. Cir. and Sys. for Video Technol. 18*, 8 (Aug. 2008), 1075–1084.

[51] Nielsen, H. B., and Madsen, K. *Introduction to Optimization and Data Fitting*. Informatics and Mathematical Modelling, Technical University of Denmark, DTU, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, aug 2010.

[52] Noorit, N., Suvonvorn, N., and Karnchanadecha, M. Model-based human action recognition. In *Second International Conference on Digital Image Processing* (feb 2010), SPIE-Intl Soc Optical Eng.

[53] Oikonomidis, I., Kyriazis, N., and Argyros, A. Efficient model-based 3d tracking of hand articulations using kinect. In *Procedings of the British Machine Vision Conference 2011* (2011), British Machine Vision Association and Society for Pattern Recognition.

[54] Palmer, F. B. Strategies for the early diagnosis of cerebral palsy. *The Journal of Pediatrics 145*, 2 (aug 2004), S8–S11.

[55] Panagiotakis, C., and Tziritas, G. Recognition and tracking of the members of a moving human body. pp. 86–98.

[56] Papadopoulos, G. T., Axenopoulos, A., and Daras, P. Real-time skeleton-tracking-based human action recognition using kinect data. In *MultiMedia Modeling*. Springer Science + Business Media, 2014, pp. 473–483.

[57] Papageorgiou, C., and Poggio, T. A trainable system for object detection. *International Journal of Computer Vision 38*, 1 (2000), 15–33.

[58] Park, J., Park, S., and Aggarwal, J. Model-based human motion capture from monocular video sequences. In *Computer and Information Sciences - ISCIS 2003*, A. Yazici and C. Sener, Eds., vol. 2869 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2003, pp. 405–412.

[59] Philippi, H., Karch, D., Kang, K.-S., Wochner, K., Pietz, J., Dickhaus, H., and Hadders-Algra, M. Computer-based analysis of general movements reveals stereotypies predicting cerebral palsy. *Dev Med Child Neurol 56*, 10 (may 2014), 960–967.

[60] Piccardi, M. Background subtraction techniques: a review. In *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)* (2004), Institute of Electrical & Electronics Engineers (IEEE).

[61] Plagemann, C., Ganapathi, V., Koller, D., and Thrun, S. Real-time identification and localization of body parts from depth images. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on* (2010), pp. 3108–3113.

[62] Pons-Moll, G., and Rosenhahn, B. Ball joints for marker-less human motion capture. In *IEEE Workshop on Applications of Computer Vision (WACV)* (2009).

[63] Portal, R., Dias, J. a., and de Sousa, L. Contact detection between convex superquadric surfaces. *Archive of Mechanical Engineering LVII*, 2 (2010), 165–186.

[64] Prechtl, H. F., and Hopkins, B. Developmental transformations of spontaneous movements in early infancy. *Early Hum Dev 14*, 3-4 (1986), 233–8.

[65] Qiao, M., Cheng, J., and Zhao, W. Model-based human pose estimation with hierarchical ICP from single depth images. In *Lecture Notes in Electrical Engineering*. Springer Science + Business Media, 2011, pp. 27–35.

[66] Rahmanpour, P. Features for movement based prediction of cerebral palsy, 2009.

[67] Rahmati, H., Aamo, O. M., Stavdahl, O., Dragon, R., and Adde, L. Video-based early cerebral palsy prediction using motion segmentation. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (aug 2014), Institute of Electrical & Electronics Engineers (IEEE).

[68] Rusinkiewicz, S., and Levoy, M. Efficient variants of the ICP algorithm. In *Proceedings Third International Conference on 3-D Digital Imaging and Modeling* (2001), Institute of Electrical & Electronics Engineers (IEEE).

[69] Sandau, M., Heimbürger, R., Villa, C., Jensen, K., Moeslund, T., Aanæs, H., Alkjær, T., and Simonsen, E. New equations to calculate 3d joint centres in the lower extremities. *Medical Engineering & Physics* (2015).

[70] Schwarz, L. A., Mkhitaryan, A., Mateus, D., and Navab, N. Estimating human 3d pose from time-of-flight images based on geodesic distances and optical flow. In *FG* (2011), IEEE, pp. 700–706.

[71] Seeley, R., Tate, P., and Stephens, T. *Anatomy and Physiology*. McGraw-Hill Higher Education, 2007.

[72] Sgandurra, G., Bartalena, L., Cioni, G., Greisen, G., Herskind, A., Inguaggiato, E., Lorentzen, J., Nielsen, J., Sicola, E., and CareToy Consortium. Home-based, early intervention with mechatronic toys for preterm infants at risk of neurodevelopmental disorders (caretoy): a rct protocol. *B M C Pediatrics 14*, 1 (2014). CURIS 2014 NEXS 326.

[73] Sharp, T., Wei, Y., Freedman, D., Kohli, P., Krupka, E., Fitzgibbon, A., Izadi, S., Keskin, C., Robertson, D., Taylor, J., Shotton, J., Kim, D., Rhemann, C., Leichter, I., and Vinnikov, A.

Accurate, robust, and flexible real-time hand tracking. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15* (2015), Association for Computing Machinery (ACM).

[74] Shen, S., Tong, M., Deng, H., Liu, Y., Wu, X., Wakabayashi, K., and Koike, H. Model based human motion tracking using probability evolutionary algorithm. *Pattern Recognition Letters 29*, 13 (2008), 1877–1886.

[75] Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., and Blake, A. Real-time human pose recognition in parts from single depth images. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition* (Washington, DC, USA, 2011), CVPR '11, IEEE Computer Society, pp. 1297–1304.

[76] Siddiqui, M., and Medioni, G. Robust real-time upper body limb detection and tracking. In *Proceedings of the 4th ACM international workshop on Video surveillance and sensor networks* (New York, NY, USA, 2006), VSSN '06, ACM, pp. 53–60.

[77] Siddiqui, M., and Medioni, G. Human pose estimation from a single view point, real-time range sensor. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops* (jun 2010), Institute of Electrical & Electronics Engineers (IEEE).

[78] Stahl, A., Schellewald, C., Stavdahl, O., Aamo, O. M., Adde, L., and Kirkerod, H. An optical flow-based method to predict infantile cerebral palsy. *IEEE Transactions on Neural Systems and Rehabilitation Engineering 20*, 4 (2012), 605–614.

[79] Stoll, C., Hasler, N., Gall, J., Seidel, H.-P., and Theobalt, C. Fast articulated motion tracking using a sums of gaussians body model. In *2011 International Conference on Computer Vision* (nov 2011), Institute of Electrical & Electronics Engineers (IEEE).

[80] Straka, M., Hauswiesner, S., Rüther, M., and Bischof, H. Skeletal graph based human pose estimation in real-time. In *Procedings of the British Machine Vision Conference 2011* (2011), British Machine Vision Association and Society for Pattern Recognition.

[81] Sundaresan, A., and Chellappa, R. Multi-camera tracking of articulated human motion using motion and shape cues. In *IN ASIAN CONFERENCE ON COMPUTERVISION* (2006), Springer, pp. 131–140.

[82] Theobalt, C., Seidel, H., Kim, K. I., Hasler, N., Stoll, C., and Elhayek, A. Spatio-temporal motion tracking with unsynchronized cameras. *2012 IEEE Conference on Computer Vision and Pattern Recognition 0* (2012), 1870–1877.

[83] Toshev, A., and Szegedy, C. DeepPose: Human pose estimation via deep neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition* (jun 2014), Institute of Electrical & Electronics Engineers (IEEE).

[84] Uldall, P., Michelsen, S. I., Topp, M., and Madsen, M. The danish cerebral palsy registry. a registry on a specific impairment. *Dan Med Bull 48*, 3 (Aug. 2001), 161–163.

[85] Vedula, S., Rander, P., Collins, R., and Kanade, T. Three-dimensional scene flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence 27*, 3 (mar 2005), 475–480.

[86] Viola, P., and Jones, M. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001* (2001), Institute of Electrical & Electronics Engineers (IEEE).

[87] Vondrak, M., Sigal, L., and Jenkins, O. C. Physical simulation for probabilistic motion tracking. In *CVPR* (2008), IEEE Computer Society.

[88] Wüst, M. Early interventions and infant health: Evidence from the danish home visiting program. *Labour Economics 19*, 4 (aug 2012), 484–495.

[89] Ye, M., Wang, X., Yang, R., Ren, L., and Pollefeys, M. Accurate 3d pose estimation from a single depth image. In *Proceedings of the 2011 International Conference on Computer Vision* (Washington, DC, USA, 2011), ICCV '11, IEEE Computer Society, pp. 731–738.

[90] Yun, K., Honorio, J., Chattopadhyay, D., Berg, T. L., and Samaras, D. Two-person interaction detection using body-pose features and multiple instance learning. In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops* (jun 2012), Institute of Electrical & Electronics Engineers (IEEE).

[91] Zhang, L., Sturm, J., Cremers, D., and Lee, D. Real-time human motion tracking using multiple depth cameras. In *Proc. of the International Conference on Intelligent Robot Systems (IROS)* (Oct. 2012).

[92] Zhou, F., la Torre, F. D., and Hodgins, J. K. Hierarchical aligned cluster analysis for temporal clustering of human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence 35*, 3 (mar 2013), 582–596.

[93] Zhu, Y., and Fujimura, K. A bayesian framework for human body pose tracking from depth image sequences. *Sensors 10*, 5 (may 2010), 5280–5293.